M 2014

U.PORTO
FEUP FACULDADE DE ENGENHARIA
UNIVERSIDADE DO PORTO

LOCALIZAÇÃO DE PESSOAS EM CENÁRIOS DE ASSISTED LIVING

ANTÓNIO GIL MOREIRA SÁ COELHO

DISSERTAÇÃO DE MESTRADO APRESENTADA
À FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO EM
MESTRADO INTEGRADO EM ENGENHARIA ELECTROTÉCNICA E DE COMPUTADORES

U.PORTO
FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO

MIEEC - MESTRADO INTEGRADO EM ENGENHARIA ELETROTÉCNICA E DE COMPUTADORES 2013/2014

A Dissertação intitulada

"Localização de Pessoas em Cenário de Assisted Living"

foi aprovada em provas realizadas em 18-07-2014

o júri

Presidente Professor Doutor Paulo José Cerqueira Gomes da Costa
Professor Auxiliar do Departamento de Engenharia Eletrotécnica e de Computadores da Faculdade de Engenharia da Universidade do Porto

Professor Doutor Valter Filipe Miranda Castelão da Silva
Professor Adjunto da Escola Superior de Tecnologia e Gestão de Águeda da Universidade de Aveiro

Professor Doutor Luís António Pereira de Meneses Corte-Real
Professor Associado do Departamento de Engenharia Eletrotécnica e de Computadores da Faculdade de Engenharia da Universidade do Porto

O autor declara que a presente dissertação (ou relatório de projeto) é da sua exclusiva autoria e foi escrita sem qualquer apoio externo não explicitamente autorizado. Os resultados, ideias, parágrafos, ou outros extratos tomados de ou inspirados em trabalhos de outros autores, e demais referências bibliográficas usadas, são corretamente citados.

Autor - António Gil Moreira Sá Coelho

Faculdade de Engenharia da Universidade do Porto

FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Localização de pessoas em cenários de *Assisted Living*

António Gil Moreira Sá Coelho

Mestrado Integrado em Engenharia Eletrotécnica e de Computadores

Orientador: Professor Doutor Luís Corte-Real

Coorientador: Doutor Pedro Carvalho

30 de Julho de 2014

Resumo

A evolução da tecnologia e da medicina tem aumentado a esperança média de vida, e consequentemente a longevidade do Homem. Apesar de todo o desenvolvimento realizado, o envelhecimento continua como um processo irreversível e as nossas capacidades vão sofrendo desgaste, dificultando a habilidade de efetuar simples tarefas. Estes problemas têm motivado os investigadores a criar soluções e mecanismos para diminuir o impacto do envelhecimento na vida das pessoas. Neste âmbito nasce o conceito de Ambiente Assistido (*Assisted living*), que consiste em casas ou espaços com sistemas inteligentes para assistir pessoas idosas ou portadoras de deficiência. Ambiente Assistido tenta combater vários problemas como a deteção de pessoas em situações de perigo, utilizando como ferramenta de resolução desses problemas a visão do computador.

Nesta dissertação é apresentado o estudo e implementação de um sistema para deteção de pessoas em cenário de Ambiente Assistido com algum nível de ocultação. Com o trabalho desenvolvido procurou-se avaliar as potencialidades da plataforma *Kinect for Windows* como base para o desenvolvimento de um sistema de deteção de pessoas em Ambiente Assistido. Foi criado um programa de aquisição de imagens utilizando o *Kinect* como plataforma de captura de imagens e realizado um conjunto de capturas de sequências para teste do algoritmo e equipamento. Foram realizadas análises às características da câmara *Kinect*, bem como às imagens por ela captadas e ao ruído de profundidade. Foi também realizado um estudo e revisão de algumas técnicas já implementadas. Por fim foi pensado e desenvolvido um algoritmo que combina a informação de profundidade e imagem tradicional do *Kinect* de forma a efetuar a deteção de pessoas mesmo com algum nível de ocultação do corpo.

O algoritmo desenvolvido tem como base Subtração de Fundo, utilizando a ideia de *Sensor Fusion* que consiste na junção de informação de dois ou mais sensores de forma a obter melhores resultados, neste caso a câmara tradicional e câmara de infra-vermelhos do *Kinect*. A subtração de fundo elimina áreas da imagem que não apresentam possíveis pessoas, admitindo que o sistema se inicia sem pessoas dentro da sala a controlar. As áreas onde existe a possibilidade de existir pessoas são analisadas com detetores de pessoas já existentes e estudados na revisão bibliográfica. Como forma de avaliação dos dados foram utilizados conjuntos de imagens marcadas manualmente para serem comparados com as marcações do algoritmo.

Abstract

The evolution of technology and medicine has increased the median life hope, increasing the man longevity. In spite of all the development achieved, the ageing continues to be an irreversible process; our skills decay in time making hard to perform simple tasks. This difficulties have motivated researchers to create solutions and mechanisms to diminish the impact of the difficulties in people's lives. In this thematics, birth the concept Assisted living, consisting in houses with smart systems to assist elder people or deficiency carrier. Assisted Living tries to fight several problems like detect people in danger situations. As tool to try to solve some of those problems is use Computed Vision.

In this project is presented a study and implementation of a system capable to perform people detection in Assisted Living environment with some occlusion level. In the work development was searched and evaluate the potential of the Kinect for Windows platform as base to development of the system of people detection in Assisted Living environment. Was created a program to capture image using Kinect as platform of image capturing and capture a group of images for test the algorithm and equipment. Was performed a study of Kinect characteristics, as well a study of the images captured by it and the noise in the depth image. Performed a study and review of some techniques already implemented. Was planned and developed a algorithm to combine the depth information and the RGB information of Kinect to perform people detection with some occlusion of body part.

The algorithm was created with the same principle as the Background Subtraction, using the Sensor Fusion that consist in using information of two or more sensors to obtain better results, in this case the fusion of RGB information and depth information. The Background Subtraction system erase areas of the image that doesn't contain people or moving objects, considering that the system was no people in the room when the system is initializing. The areas where was the chance of existing people is processed and classified with people detectors already studied in the technical review. As way to evaluate the data was used a set of images marked by hand to be compared with the images marked by the algorithm.

Agradecimentos

As minhas primeiras palavras de apreço têm que ir, sem qualquer dúvida, para os meus Pais e Irmão. Eles são a razão da minha existência, o meu porto seguro, a minha luz guia, fonte de inspiração e força. Sem eles não seria quem sou, nem estaria aqui hoje no final desta etapa. A eles devo tudo o que tenho e o que venha a ter. Um agradecimento especial a toda a minha família pelo seu apoio.

Quero prestar também agradecimento ao meu Orientador Professor Doutor Luís Corte-Real e ao meu Coorientador Externo Doutor Pedro Carvalho pela supervisão prestada durante todas as fases do projecto, pelas reuniões de aconselhamento e verificação do trabalho e por ter sido guia na execução do trabalho de forma mais correta.

Uma palavra de agradecimento aos colegas de trabalho do INESC que me ajudaram na execução do trabalho, me deram apoio, ideias e ajuda técnica.

Não posso deixar de agradecer aos meus amigos. Aos da FEUP, que representam a minha "família de engenheiros". Os amigos que me acompanharam ao longo destes 5 longos anos de trabalho árduo, de muitas horas de estudo, de muitos risos e bons momentos e partilha de todas as experiências que este curso e faculdade me facilitou. Aos amigos da UTAD, que por muito efémero que fosse o tempo que estive com eles, foram uma grande influência na minha caminhada e foi um enorme gosto tê-los conhecido. Aos amigos da minha terra, que me acompanharam ao longo de todo este percurso, que me apoiaram, que me deram força e partilharam comigo bons e maus momentos ao longo destes anos. E aos demais amigos que passaram pela minha vida académica e pessoal.

A todas as outras pessoas que passaram pela minha vida e deixaram marca de alguma forma um obrigado.

Muito Obrigado a todos.

António Gil Coelho

"The true sign of intelligence is not knowledge but imagination."

Albert Einstein

Conteúdo

Lista de Figuras

Lista de Tabelas

Abreviaturas e Símbolos

AAL	*Ambient Assisted Living*
ANN	*Artificial Neural Networks*
API	*Application Programming Interface*
CPU	*Central Processing Unit*
FPS	*Frames per Secound*
GPU	*Graphics Processing Unit*
HCI	*Human Computer Interface*
HD	*High Definition*
HOD	*Histogram of Oriented Depths*
HOG	*Histogram of Oriented Gradients*
IR	*Infra-Red*
LIDAR	*LIght Detection And Ranging*
LBP	*Local Binary Patterns*
MOG	*Mixture of Gaussian*
PC	*Personal Computer*
PCL	*Point Cloud Library*
RADAR	*RAdio Detection And Ranging*
RGB	*Red Green Blue*
RGB-D	*Red Green Blue - Distance*
ROI	*Regions of Interest*
SDK	*Software Development Kit*
SIFT	*Scale-Invariant Feature Transform*
SVM	*Support Vector Machine*
TOF	*Time of flight*
VTR	*Video Tape Recorder*
XML	*eXtensible Markup Language*

cm	centímetros
m	metros

Capítulo 1

Introdução

1.1 Motivação

O envelhecimento populacional é um fenómeno a nível mundial que ainda não apresenta solução. A vida, cada vez mais prolongada tem levantado questões e preocupações na procura de garantir qualidade e vitalidade para a terceira idade. Com o intuito de aumentar a esperança média de vida, têm-se gerado grandes projetos na área da medicina acompanhados pelos avanços da tecnologia. Apesar de todo o desenvolvimento conseguido, o envelhecimento continua como um processo irreversível e as nossas capacidades vão sofrendo desgaste, dificultando a habilidade de efetuar simples tarefas. Estas dificuldades têm motivado invenções que possam garantir uma vida com qualidade e os investigadores esforçam-se para criar soluções que permitam a autonomia de qualquer indivíduo. A população idosa representa uma grande percentagem de pessoas com problemas de saúde, mobilidade e dificuldade em trabalhar, fatores derivados da idade avançada. Estes problemas representam grandes custos e encargos para os familiares e para o país. Tendo em conta os dados demográficos apontados pela *HelpAge*, estima-se que em 2050, nos países desenvolvidos, uma em cada cinco pessoas tenha idade superior a 60 anos [1]. No caso de Portugal 24.4% dos 10.6 milhões de habitantes tem idade superior a 65 anos - dados registados em 2013 [2]

"O mundo está a envelhecer e nós precisamos estar preparados para isso" *HelpAge* [1].

Sendo a Engenharia uma arte e ciência com um conjunto de técnicas para a criação de máquinas e sistemas com o principal intuito de trazer benefício e conforto ao ser humano, é utilizada por investigadores e empresas na tentativa de criar soluções que permitam às pessoas idosas ter melhor qualidade de vida, segurança e uma participação mais ativa na sociedade [3].

Nesta temática nasceu um conceito chamado Ambiente Assistido (*Ambient Assisted living*) que consiste em casas desenhadas e especialmente destinadas para acomodar pessoas idosas ou portadoras de alguma deficiência. Para tornar as suas vidas mais confortáveis, estas são devidamente equipadas com aparelhos que permitam maior segurança através de supervisão e assistência nas atividades diárias de forma a assegurar o bem estar dos que nelas habitam.

Segundo a *Ambient Assisted Living* (AAL) [4], o Ambiente Assistido tem como principais objetivos a criação de melhores condições de vida para idosos, criando sistemas para *"aging well"* aplicados nas suas casas aumentando assim a qualidade de vida e à redução dos custos de cuidados de saúde [4].

Os objetivos da AAL passam por: aumentar o tempo que as pessoas podem viver no seu ambiente preferido (casa), aumentando a sua autonomia, mobilidade e privacidade; manutenção da saúde e autonomia dos idosos; proporcionar uma vida mais saudável para idosos em risco; fornecer segurança a pessoas isoladas e apoiar carreiras profissionais de familiares que tomam conta de idosos.

Entre os vários problemas que o Ambiente Assistido tenta combater, é de se realçar o combate à solidão, o aumento de segurança e vigilância para deteção de intrusos mas principalmente para a deteção de situações de perigo ou problemas com as pessoas que se encontram na casa. Ambiente Assistido pretende utilizar vídeo de forma a dotar o sistema de capacidades de análise do ambiente, semelhantes à visão humana. Esta capacidade permitirá a deteção de pessoas e situações de perigo através da análise do cenário e dos seus constituintes, interpretação e classificação dos elementos ou acontecimentos tal como as pessoas fazem ao visionar uma sala.

Apesar da tecnologia existente já ser capaz de operar em certas condições e produzir bons resultados a nível de deteções e seguimento, ainda existem problemas a resolver em certos cenários de Ambiente Assistido tais como a deteção de pessoas em divisões com elevada probabilidade de ocultação de alguma parte do corpo, o que dificulta a deteção pelas técnicas existentes. Na procura de resolução destes problemas nasce esta dissertação que tem como principal objetivo o estudo e implementação de um sistema capaz de efetuar deteção de pessoas em cenário de Ambiente Assistido com algum nível de ocultação.

1.2 Objetivos

Nesta dissertação é proposto o estudo e desenvolvimento de um módulo de software para deteção de pessoas utilizando o equipamento *Kinect for Windows* (apresentado em 3.1) como alternativa a câmaras convencionais e outras soluções. Pretende-se detetar pessoas com algum nível de ocultação do corpo para que seja possível determinar a sua localização no espaço 3D de uma sala, procurando identificar as vantagens e desvantagens em relação a câmaras RGB convencionais. Realizar uma análise das caraterísticas do equipamento, a possibilidade de segmentação da imagem utilizando a informação RGB e de profundidade e a influência/imunidade a variações de iluminação.

Aqui será feita uma análise de algumas soluções existentes tal como é o caso do equipamento a usar, o *Kinect*, que segue a ideia de *sensor fusion* combinando informação da câmara RGB e sensor de distância que possui para fazer melhores medições.

1.3 Estrutura do Documento

Este documento está dividido em 6 capítulos.

No Capítulo 2 intitulado de Revisão de Literatura são apresentadas soluções estudadas e existentes no mercado, técnicas e uma breve referência à historia de alguns termos e equipamentos relacionados com o tema.

No Capítulo 3 é apresentada e discutida a Plataforma Experimental utilizada na Dissertação. É apresentado o equipamento utilizado, alguns dos testes efetuados com o *Kinect* para determinação de algumas limitações e características, sequências capturadas para testes e métricas de avaliação.

Após a apresentação da Plataforma surge o Capítulo 4 onde é apresentada uma proposta de algoritmo para deteção de pessoas com fusão de informação.

Em penúltimo lugar temos o Capítulo 5 onde são apresentados Resultados. Neste capítulo são apresentados resultados dos testes efetuados com os detetores e uma comparação entre os resultados de subtração de fundo usados.

Por ultimo temos o Capítulo 6 onde são apresentadas conclusões e ideias para trabalho futuro. Apresentam-se ideias e propostas de resolução de alguns problemas encontrados ou soluções para melhorar certas partes do trabalho bem como conclusões de um modo geral e em alto nível.

Em anexos estão presentes imagens referentes ao diagrama de classes do programa criado para executar o algoritmo, bem como fluxogramas representativos dos algoritmos e sequências de capturas.

Capítulo 2

Revisão Bibliográfica

Neste capítulo serão abordadas técnicas e temáticas referidas na bibliografia revista. Na primeira secção deste capítulo será apresentada uma referência a alguns conceitos relativos ao tema. Na segunda secção serão apresentadas técnicas e métodos estudados para a deteção de pessoas através de equipamentos como o *Kinect*, câmaras convencionais e infra-vermelho. De inicio é apresentada uma breve referência histórica a equipamentos relacionados com o trabalho apresentado.

O primeiro gravador de vídeo (*Video Tape Recorder* VTR) foi desenvolvido em 1951 com a capacidade de gravar imagens de câmaras de televisão a partir da sua conversão em impulsos elétricos e a sua gravação em fita magnética [5]. Durante os anos 60 a NASA contribuiu para o desenvolvimento das câmaras digitais devido à investigação espacial. Nessa altura, a NASA estava a transitar do uso do sinal analógico para sinal digital nos satélites usados para mapear a superfície da lua. Também nos anos 60 o governo dos Estados Unidos da América contribuía para o desenvolvimento da tecnologia de imagem digital pelo uso desta para fins militares em satélites espiões.

Em 1972, o setor privado entrou na história da câmara digital com a primeira patente por parte da Texas Instruments sobre a primeira câmara eletrónica sem filme tradicional. Durante os anos 70, a Kodak também contribuiu para a evolução da câmara digital inventando vários sensores para a captação de imagem digital, seguindo-se a Sony com o lançamento da primeira câmara comercial no início dos anos 80. As primeiras câmaras a serem postas à venda para o público em geral foram a Apple QuickTake 100 em 1994, seguida de câmaras tais como a Kodak DC40 e a Casio QV-11 produzidas em 1995 e a Sony's Cyber-Shot Digital Still Camera em 1996 [5].

Desde então a câmara digital tem evoluído e tem sido usada nas mais diversas situações, muitas das quais transcendem os fins lúdicos e comerciais.

Após o nascimento do laser, iniciaram-se as tentativas de aliar este ao conhecimento da tecnologia radar de forma a possibilitar medições com laser. No início de 1960 começou a aparecer equipamento com a capacidade de medir distâncias a partir de laser. A sua aplicação inicial foi no âmbito da meteorologia através de medições do tamanho de nuvens. Uma das aplicações mais conhecidas foi na missão *Apollo 15*, no ano 1971, onde esta foi usada para mapear a superfície

da lua. Esta tecnologia foi apelidada de *LIDAR (LIght Detection And Ranging)* à semelhança de *RADAR (RAdio Detection And Ranging)* pois ambas se apoiam na mesma forma de medir a distância, usando apenas ondas diferentes para o efeito: *LIDAR* por laser e o *RADAR* por ondas rádio [6]. Com a tecnologia *LIDAR* e com a evolução dos processadores foi possível, por volta do ano 2000, o aparecimento das câmaras *Time of flight* (TOF) à venda para o público em geral. Desde então, estas têm sido usadas em aplicações na área da robótica, setor automóvel e em sistemas de medições [7].

O *Kinect* foi lançado a 4 de novembro de 2010 nos Estados Unidos da América, sendo este o primeiro país a receber este acessório da consola *XBox 360*, desenvolvido pela *Microsoft* em parceria com a *PrimeSense*. O sensor de profundidade usado pelo *Kinect* foi desenvolvido em 2005, sendo anunciado oficialmente em 1 de junho de 2009 com o nome de *"Project Natal"* [8].

Como uma das funções inicialmente pensadas para o *Kinect*, estava a capacidade de usar gestos e comandos de voz para interagir com a *XBox 360*. Esta função foi uma das razões para a escolha das especificações do equipamento tal como a capacidade de construir um "esqueleto" do utilizador através dos dados fornecidos a partir do sensor de profundidade.

Com o anúncio e lançamento do *Kinect* foram também lançados e apresentados jogos que utilizavam as caraterísticas deste, principalmente a de desenhar o "esqueleto" do utilizador. Alguns destes jogos apresentavam uma interação com o utilizador que incluíam pintar, interagir com a personagem do jogo e até mesmo praticar desportos. Estes jogos vieram mostrar a evolução da tecnologia apresentada pelo *Kinect* que permitia ao utilizador interagir com um computador sem o uso dos periféricos comuns, tornando o seu corpo num comando e desta forma estendendo as técnicas de *Human Computer Interface* (HCI) tradicionais.

2.1 Equipamento

Nesta secção são apresentados equipamentos considerados para a realização da dissertação. Na apresentação de câmaras semelhantes ao *Kinect*, é feito uma comparação entre elas de forma a analisar as vantagens e desvantagens entre elas.

2.1.1 Conceitos

De inicio são apresentados alguns conceitos relacionados com imagem tradicional e imagem de profundidade, bem como alguns equipamentos que produzem essas imagens.

- **Câmaras Stereo:**

Equipamento com duas óticas ou constituído por duas câmaras dispostas a par a uma distância que permite simular a visão dos olhos de um Humano, como exemplo figura 2.1. Com o uso deste tipo de câmaras é possível calcular a distância da câmara a um objeto utilizando apenas imagem RGB.

Figura 2.1: Exemplo de Câmara *Stereo PointGrey Bumblebee2* (extraída de [9])

- **Câmaras *Time of Flight* (TOF):**

Equipamento que produz imagens de profundidade, ou seja, produz imagens ou dados relativamente à distância de objetos posicionados à frente da câmara (figura 2.2). Para tal, estes equipamentos usam o conhecimento da velocidade da luz e produzem medições de distância através da iluminação de um objeto por um feixe laser e pela análise da luz refletida pelo objeto.

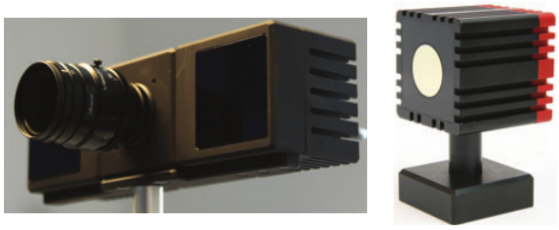


Figura 2.2: Exemplo de Câmaras TOF (adaptada de [7])

- **Imagem multiespetral:**

Imagem constituída por imagens de diferentes comprimentos de onda, por exemplo, imagens de luz visível, infra-vermelho e ultravioleta.

- **Câmaras convencionais:**

Equipamento capaz de captar imagens com três canais de cor, vermelho (R), verde (G) e azul (B). Exemplos deste tipo de equipamento são as câmaras tradicionais de fotografia ou vídeo usadas hoje em dia.

- **Imagem RGB-D:**

Imagens constituídas por 4 canais, nos quais os 3 primeiros correspondem a uma imagem RGB convencional e o último canal representa o valor de distância, para cada píxel, à câmara que produziu a imagem (figura 2.3). Estas imagens são produzidas por equipamentos que usam *sensor fusion*, tal como o *Kinect*. Estas imagens entram no conceito de *Sensor Fusion* que consiste na

Figura 2.3: A imagem da esquerda representa os 3 canais de RGB e a da direita a imagem de distância produzida pelo *Kinect*.

combinação de dados fornecidos por vários sensores, geralmente, de grandezas diferentes para produzir medições ou deteções com maior precisão do que usando apenas um tipo de sensores.

Como equipamento capaz de produzir imagens RGB-D temos, por exemplo o *Kinect*.

O *Kinect* é uma câmara desenvolvida pela *Microsoft* e *PrimeSense* como acessório para a consola *Xbox 360* capaz de produzir imagens RGB-D. É constituída por uma câmara RGB, um conjunto emissor recetor infra-vermelho que permite obter um valor de distância entre 40cm e 4m com alguma precisão para cada píxel da imagem, por um conjunto de 4 microfones que permitem ter uma noção da localização da origem do som devido à sua disposição no equipamento e por um conjunto de motor e acelerómetro [10, 11] (figura 2.4). Esta câmara tem sido inúmeras vezes usada em investigação e equipamento robótico, pelas suas capacidades, software já desenvolvido e sobretudo preço. Entre algumas das aplicações está o uso do *Kinect* como sistema de sensores para determinar se existem pessoas em frente do robô e segui-las [12] e deteção de pessoas utilizando apenas a informação de profundidade [13], entre outras.

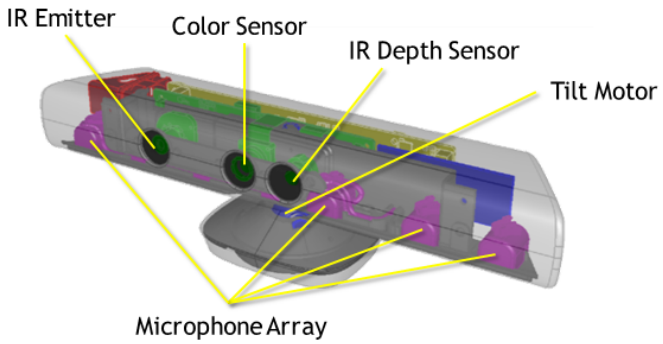


Figura 2.4: Disposição dos sensores do *Kinect*

Todos os sensores podem ser utilizados para diversas aplicações. O motor tem finalidade efetuar ligeiros ajustes da posição do *Kinect*, sendo possível controlar por código e obter a inclinação do equipamento a partir de um acelerómetro.

2.1.2 Equipamentos

Nos últimos anos têm aparecido cada vez mais equipamentos e algoritmos que aplicam o conceito de *sensor fusion* com vários tipos de sensores e de informação no sentido de aumentar a precisão e robustez das medições efetuadas, e poder produzir resultados que, recorrendo apenas a um tipo de sensor não seriam possíveis. Nas câmaras e em imagem também existe essa tendência: vários conjuntos de diferentes sensores aliados a câmaras de captação de imagem RGB ou na escala de cinzento.

Para a execução desta dissertação procuramos identificar câmaras RGB com sensores de distância. Estes equipamentos têm vindo a aparecer cada vez mais devido à evolução da velocidade de cálculo e da necessidade de melhorar e dotar computadores de capacidades de visão avançadas.

Atualmente no mercado existem várias câmaras com sensores TOF das quais sobressaem a câmara da *Microsoft* (*Kinect for Windows*) [10] (figura 2.5), da *Asus* (*Xtion PRO*) [14](figura 2.6) e *SoftKinetic* (*DepthSense 311*) [15](figura 2.7) e câmaras stereo como a *Bumblebee2*(figura 2.1) da *Point Grey* com preços na gama dos 1400 €[9, 16]. Na tabela 2.1 vemos a comparação entre três das câmaras referidas anteriormente.

Tabela 2.1: Tabela de dados comparativos entre as câmaras.

	Kinect	*Xtion PRO*	*Store DS311*
Marca:	*Microsoft*	*Asus*	*SoftKinetic*
Preço:	+/- 150€	+/- 170€	+/- 220€
Sensores: Câmara RGB Sensor de profundidade Microfones	 Sim Sim 4	 Sim Sim 2	 Sim Curto alcance (15cm-100cm) Longo alcance (1.5m-4m) 2
Software:	*Kinect for Windows SDK* *OpenNI* *OpenKinect*	*OpenNI*	Sem informação

Figura 2.5: *Kinect for Windows* (extraída de [10]).

Figura 2.6: *Asus Xtion PRO* (adaptada de [14]).

Figura 2.7: *SoftKinetic Store DS311* (adaptada de [15]).

2.2 Técnicas

2.2.1 Deteção de Humanos com imagem térmica

Com o estudo de deteção de pessoas recorrendo a imagens RGB foi constatado que é um ambiente difícil de efetuar boas deteções pelas semelhanças de cor entre a pessoa e o ambiente em seu redor. Por esse motivo, e de forma a melhorar a deteção, foi considerado e estudado o uso de imagens nas quais exista informação que ajude a separar a pessoa do fundo tais como imagens térmicas.

O'Malley et al. [17] demonstram uma forma de deteção de pessoas utilizando imagens térmicas para deteção num ambiente noturno. Esta investigação partiu da tentativa de diminuir o número de atropelamentos na estrada devido à má visibilidade dos pedestres por parte dos condutores.

Para a deteção de pedestres com imagens de infra-vermelho podem ser aplicados alguns dos métodos usados em imagem RGB. Apesar de diferenças fundamentais na informação das imagens, é possível o uso de alguns métodos diretamente ou com adaptações para tratar a informação. Neste caso foi utilizada como primeiro passo uma técnica que envolve o método de crescimento de regiões (*region growing*), tendo como início a deteção de regiões de interesse (*Regions of Interest* - ROI) para depois se efetuar uma análise e classificar objetos como "pedestre"ou "não pedestre". O método de *region growing* tem uma abordagem iterativa para a segmentação de objetos a partir da agregação de pontos em redor de pontos iniciais (sementes). Os pontos iniciais são escolhidos utilizando um critério exigente. Por exemplo, no caso de deteção de pessoas com imagem térmica, o critério consistiu na utilização de pontos de intensidade muito elevados (correspondentes a uma temperatura alta) escolhidos de forma a obter o melhor resultado devido à existência de uma temperatura elevada do corpo humano em relação ao ambiente. Uma vez determinados os pontos iniciais, inicia-se o processo iterativo no qual se agrega, aos pontos da iteração anterior, os pontos da vizinhança que satisfazem uma condição menos exigente, sendo este o critério de agregação.

Na figura 2.8 ilustram-se alguns dos resultados referidos no artigo [17] onde se pode observar, rodeado a azul, a área da imagem que a experiência identificou como pessoa.

Figura 2.8: Alguns resultados da experiência efetuada por O'Malley e o seu grupo (extraída de [17]).

A classificação é efetuada a partir da comparação do vetor de caraterísticas criado pelo método *Histogram of Oriented Gradients* (HOG 2.2.5) entre a imagem a analisar e um conjunto de imagens previamente estudado. Esta comparação é efetuada por métodos de *machine learning* tais como *Support Vector Machine* (SVM) ou *Artificial Neural Networks* (ANN) que são criadas através

de um conjunto de treino constituído por imagens de teste e resultados esperados. As imagens são comparadas usando a escala de cinzento, pois tal dá origem a melhores resultados do que classificação a partir de imagens binárias devido aos classificadores binários serem demasiado sensíveis à forma do objeto.

Outra forma de deteção de pessoas com imagem térmica é apresentada por Sanoj el al. [18] onde descrevem o desenvolvimento de um sistema automático de vigilância para seguir pessoas e determinar se estes se encontram em situação de risco. Para a deteção são mencionadas quatro fases: a segmentação; *supervised learning*; determinação de cantos; e subtração de fundo.

Como forma de seguir a pessoa são referidas três formas convencionais: subtração de fundo; diferenciação temporal; fluxo ótico (*optical flow*). A técnica de subtração do fundo é utilizada para seguir a pessoa a partir da comparação entre *frames* consequentes. A diferenciação temporal deteta movimento a partir da comparação de píxel a píxel entre imagens consequentes. Fluxo ótico é o padrão de movimento aparente do objeto numa área, causado pelo movimento relativo do observador (neste caso câmara) e a área de observação [19]. O fluxo ótico é pesado a nível computacional o que traz dificuldades em utilizar este método em tempo real sem o uso de hardware adicional para ajudar o cálculo.

No artigo [20] recorre-se ao *sensor fusion* com uma câmara convencional e a uma câmara térmica, através do qual se detetam áreas de pedestres usando uma adaptação do método de subtração de fundo seguido da estimação do número de pedestres através do *head-candidate selection algorithm*. É escolhido o algoritmo referido, pois a cabeça do pedestre é a parte do corpo mais fácil de determinar devido à sua posição elevada e a probabilidade de ocultação ser menor, sendo também a parte do corpo mais evidente. Na classificação da secção da imagem analisada é usado um modelo probabilístico de inferência Bayesiana baseada em informação previamente introduzida com base em parâmetros do físico humano e do *layout* da área de deteção.

2.2.2 Deteção de pessoas com imagem de profundidade

Uma forma de deteção de pessoas utiliza a informação de distância e a deteção da forma de "ombro-cabeça-ombro"[21].

Técnicas convencionais usadas para a deteção de pessoas, tais como HOG (2.2.5) com algoritmos de *machine learning* tais como SVM [22] requerem muita informação para treinar o sistema para atuar em situação real, podendo surgir dificuldades devido ao ambiente onde vai ser implementado ser bastante diferente do usado para o treino do método.

Em resposta aos problemas apresentados, têm aparecido sistemas de deteção de pessoas com a captação da imagem de cima para baixo e sem o uso de *machine learning*. Algumas das vantagens em detetar pessoas vistas por cima, ou seja, com a câmara posicionada acima das pessoas e direcionada para o chão, são a diminuição da influência do fundo da imagem se o chão for plano e regular e na dificuldade de “esconder” uma pessoa devido à existência de algum objeto entre a câmara e a pessoa. Esta técnica já apresenta produtos no mercado como:

- **Palossie Customer Traffic Counting System:**

Sistema produzido pela *Giken Trastem*, consegue contar o número de pessoas que passam numa determinada zona através de sensores instalados no teto de entradas de lojas ou edifícios. Consegue recolher informação de pessoas a entrar ou a sair em simultâneo, e enviar essa informação de tráfego para um computador para ser analisada [23]. Conceito ilustrado na figura 2.9.

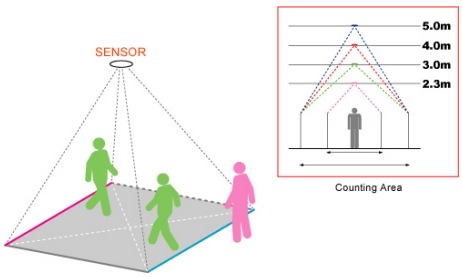


Figura 2.9: Funcionamento do sistema do *Giken Trastem* (adaptada de [23]).

- **Censys3D People Tracking System:**

Censys3D Software Development Kit (SDK) produzido pela *Point Grey* é um *software* que utiliza imagens de uma câmara *stereo* para obter informação de tráfego de pessoas numa área exterior ou interior de um edifício, sendo imune a sombras e variações da luminosidade [24]. No método proposto por Ikemura e Fujiyoshi é efetuada a deteção de pessoas recorrendo a um descritor baseado em *Haar-like* para a deteção de "ombro-cabeça-ombro". Para reduzir a informação a ser processada, são obtidas as áreas a analisar a partir de subtração de fundo. De seguida é aplicado o descritor em cada área, para depois passar a informação gerada por um algoritmo de *mean-shift clustering* para classificar a informação.

Na figura 2.10 pode se observar o modelo implementado por Ikemura e Fujiyoshi [21], em que se pode observar no canto superior esquerdo da figura os “moldes” para a deteção em que estão presentes as posições e formas que este método consegue detetar e na parte inferior direita da figura o resultado do uso deste método aplicado numa imagem.

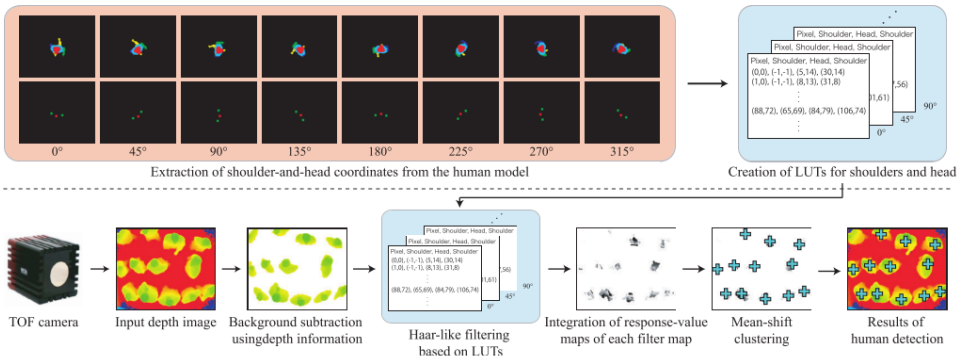


Figura 2.10: Estrutura do método *Haar-like* implementado no artigo [21].

2.2.3 Deteção de pessoas com o uso do Kinect

A deteção de pessoas com o uso de imagens de profundidade tem sido usada devido às caraterísticas destas imagens, apesar do elevado preço dos equipamentos tipo câmaras TOF que as captam. Com o aparecimento do *Kinect* tem-se assistido a uma integração desta câmara na investigação de algoritmos de deteção de pessoas e objetos com o uso de imagens de profundidade devido às suas caraterísticas e ao seu preço.

Robôs equipados com câmaras podem obter imagens do ambiente que os rodeia para identificar objetos e ajudar a determinar a localização do robô. Desta forma, a visão por computador é uma área de interesse e investigação na robótica.

Contudo, o uso de imagens 2D para representar um mundo 3D apresenta grande dificuldade e pouca eficiência para identificar e seguir formas num ambiente complexo. Devido às caraterísticas do *Kinect*, este tem vindo a ser usado como periférico em robótica, dotando o robô da capacidade de seguir pessoas num ambiente com obstáculos como paredes. Com o uso do *Kinect* e da sua capacidade de gerar imagem RGB-D e "esqueleto" de pessoas que se encontrem no seu campo de visão, a tarefa de identificação do ambiente e de pessoas ficou mais simples e eficaz.

No artigo [12] é apresentado um sistema de seguimento de pessoas para um robô recorrendo ao *Kinect*. O reconhecimento da pessoa presente à frente do robô é feito a partir da capacidade de *skeleton tracking* (seguimento do esqueleto) que o *Kinect* possui para desenhar o "esqueleto"das pessoas. A posição da pessoa relativamente ao robô é retirada a partir da informação de distância produzida pelo *Kinect*. A junção destas duas informações, sendo elas a do esqueleto e da distância, torna o reconhecimento da pessoa mais eficaz e rápido ficando a aplicação mais próxima de tempo real. Na figura 2.11 ilustra o conjunto de fotogramas que apresenta o resultado da experiência referida no artigo [12].

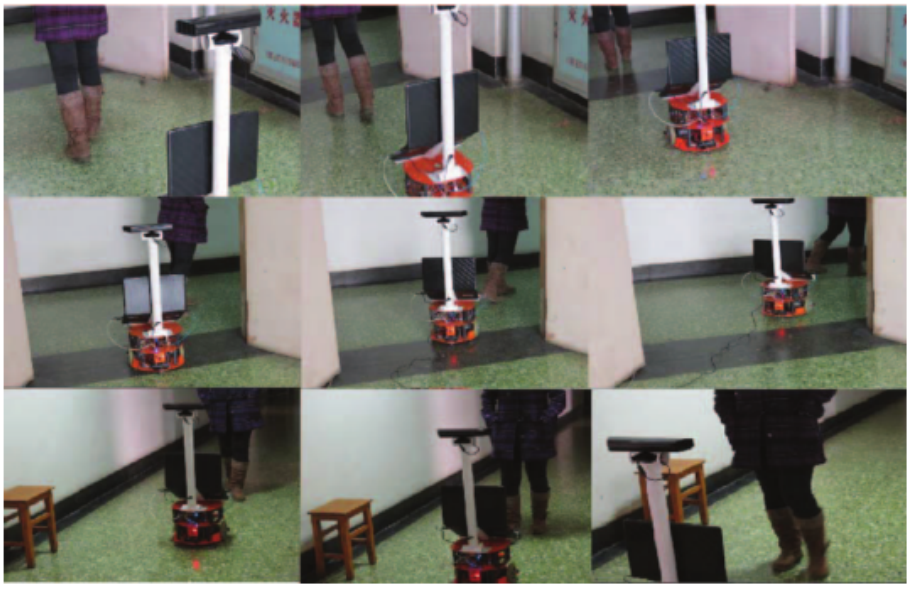


Figura 2.11: Demonstração do robô a seguir uma pessoa dentro de um corredor (adaptada de [12]).

Uma outra forma de deteção de pessoas por parte de um robô é apresentada em [25] onde o robô procura uma pessoa para lhe pedir indicações para cumprir o seu objetivo. A deteção referida é baseada na deteção de três elementos para o robô saber que está na presença de uma pessoa, sendo elas: a deteção das pernas efetuada por um *laser range finder*; a deteção de pele e a deteção da cara da pessoa. A cara é detetada a partir do detetor de face criado por Viola e Jones que apresenta semelhanças ao filtro de *Haar-Like* [26] . O detetor implementado por Viola e Jones tem como base a aplicação de uma máscara constituída por retângulos escuros e claros e o cálculo de um valor que caracteriza a zona de píxeis cobertos pela máscara através da subtração da soma dos píxeis da zona sobreposta pelo retângulo claro e da soma dos píxeis da zona sobreposta pelo retângulo escuro. Este método de deteção de pessoas é mais adaptado à deteção de uma pessoa em frente do robô e que esteja virada para o mesmo para interação, apresentando limitações e problemas em cenários que exista ocultação da cara, pernas ou zonas de pele da pessoa.

Em [13] foi proposta a deteção de pessoas através da imagem de profundidade do *Kinect for Xbox360*. O método implementado faz a deteção de pessoas através da deteção da cabeça em duas etapas seguidas de uma segmentação para detetar o resto do corpo visível. Na primeira etapa o algoritmo utiliza a informação de contornos proveniente do método *Canny edge detector* aplicado à imagem de profundidade para localizar uma região onde possa existir uma pessoa. De forma a localizar possíveis cabeças, é usado *2D Chamfer distance matching* [27] que produz regiões que são classificadas na próxima etapa como cabeça ou não. Na figura 2.12 é apresentada a primeira etapa da deteção de cabeças apresentada em [13].

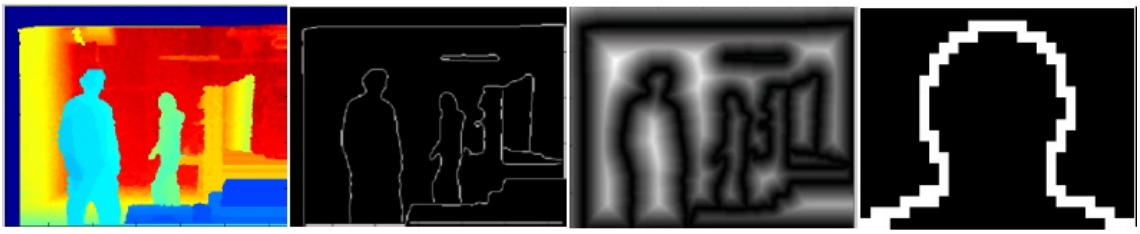


Figura 2.12: Da esquerda para a direita, imagem de profundidade, contornos, imagem de distância dos pixeis e *template* usado no *2D Chamfer distance matching* (extraída de [13]).

Na segunda etapa, é criado um modelo 3D da cabeça para verificar se a possível cabeça encontrada é na verdade uma cabeça. Para a criação do modelo 3D foi previamente estudada, a relação entre a altura da cabeça e a distância à câmara. Deste estudo foram criadas equações que permitem calcular parâmetros das dimensões da cabeça em função da distância para verificar se a área em análise é uma cabeça. Uma vez obtida a localização de uma cabeça, é determinado o contorno do corpo para efetuar a segmentação do corpo em relação à imagem. Para a segmentação é necessário filtrar a imagem primeiro.

A filtragem tem como principal objetivo realçar a separação entre os pés e o chão, porque em imagens de profundidade estes apresentam valores iguais, dificultando a segmentação do corpo em relação ao chão. A filtragem usada delimita áreas planares que são paralelas ao chão, sendo

possível determinar o contorno do corpo e separar os pés do chão. Após a separação do corpo em relação ao chão é determinado o contorno do corpo da pessoa. É usado como semente para o *Region Growing* a região produzida pela deteção de cabeça e como critério de agregação a relação entre um *pixel* e sua vizinhança. A relação usada passa pela diferença entre as distâncias medidas nos pixeis, pois é assumido que os valores de distância da superfície do corpo medidas pelo *Kinect* são contínuos e variam pouco dentro de uma gama especifica [13]. Na figura 2.13 são apresentados os resultados das experiências demonstradas no artigo [13].

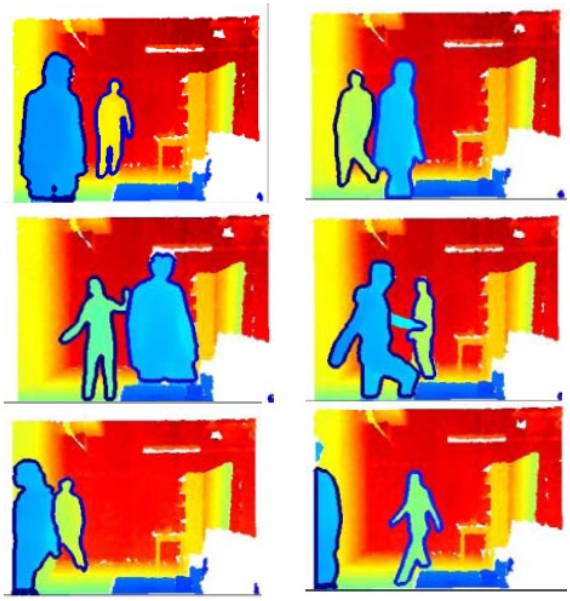


Figura 2.13: Resultados das experiências demonstrados no artigo de Lu Xia e sua equipa [13].

Em [28], Spinello e Arras também apresentam uma solução para a deteção de pessoas utilizando imagens RGB-D. A solução apresentada assenta no uso de um detetor criado pelos autores chamado *Histogram of Oriented Depths* (HOD). Este detetor é inspirado no HOG [29](apresentado em 2.2.5), utilizando a mesma ideia central do detetor, tendo sido adaptado para o uso em imagens de profundidade. Neste artigo também é proposto um detetor chamado *Combo-HOD*, um detetor que combina as probabilidades do detetor HOD e o HOG usando imagens RGB-D.

O detetor HOD segue a mesma base de funcionamento do HOG tendo como diferença o uso do gradiente de profundidade em vez do gradiente da cor. O *Combo-HOD* utiliza os dois detetores de forma a aumentar a deteção nas imagens RGB-D e utilizar o máximo de informação possível. Os detetores são usados após efetuar *scale-space search*. O *scale-space search* retorna uma área da imagem que contem um objeto encontrado com uma ou mais caraterísticas, como por exemplo um objeto com altura de 1,7 metros. Uma vez determinada uma área de interesse, são aplicados os detetores nessa área: o HOD na área correspondente da imagem de profundidade e o HOG na área da imagem RGB. De seguida a informação é fundida de forma a produzir uma melhor deteção. Quando não existe informação de profundidade, o sistema opera apenas como HOG normal. Na figura 2.14 está ilustrado os resultados apresentados no artigo [28].

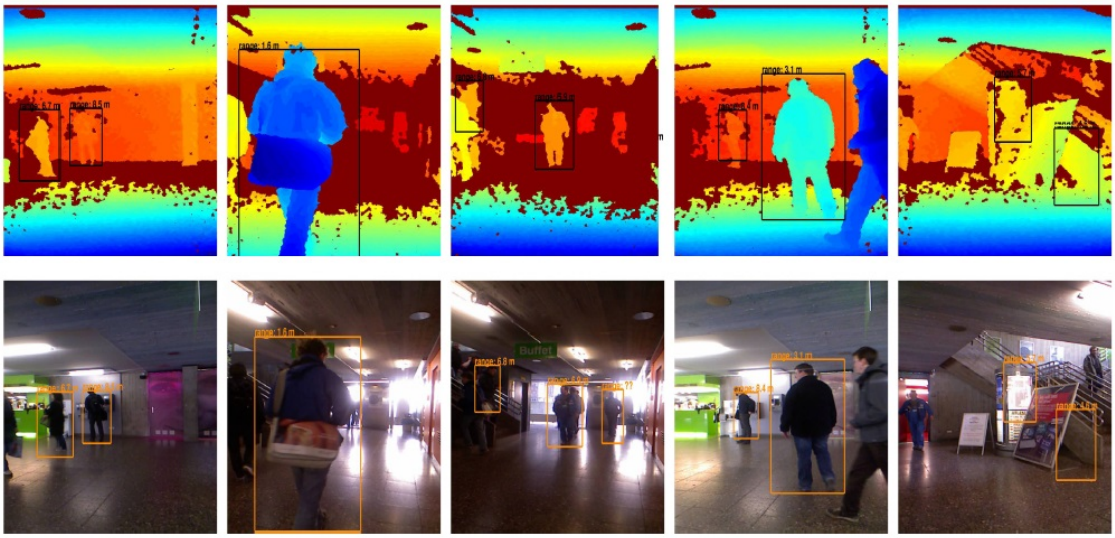


Figura 2.14: Resultados de deteção pelo detetor *Combo-HOD* apresentados em [28].

2.2.4 Detetor de Face

Uma outra forma de deteção de pessoas baseia-se na deteção de faces de pessoas. Em [30] é apresentado uma avaliação de detetor, seguimento e reconhecimento de face para aplicar em plataformas robóticas em cenários de *Assisting Living* utilizando um *Kinect*. Para o efeito de deteção de face foi usado o algoritmo de Viola e Jones com *Haar-Like* baseado em *Haar wavelets*. O algoritmo foi implementado usando o OpenCV, que possui várias combinações retangulares de *Haar wavelets* e efetua a classificação da informação usando um algoritmo *Cascade of Classifiers*. O detetor em OpenCV procura retângulos na imagem que possam conter objetos/faces que são classificados pelo método de *Cascade of Classifiers*. É usado *Camshift* para o seguimento da face. O *Camshift* usa a informação de cor para efetuar o seguimento de objetos e através do algoritmo de *mean-shift* obtém a distribuição dessas cores. Por fim cria um histograma de cor para representar a face, que se consegue adaptar dinamicamente à distribuição de probabilidades que está a seguir. Para o reconhecimento foi utilizado *Principal Component Analysis* (PCA) usando *Eigenfaces*, que é um método de extração de caraterísticas da face, posteriormente utilizado para comparar estatísticas com uma base de dados para se obter o reconhecimento.

Nos resultados apresentados (figura 2.15) e conclusões é descrito que com a imagem de resolução 640 x 480 até 1280 x 960 o algoritmo consegue detetar faces até uma distância de 83 cm, sendo o ideal entre 68 a 78 cm. Por este motivo este detetor não é o mais aplicável a esta dissertação. A distância ideal é caraterizada pela zona na qual o detetor funciona de forma contínua. O detetor mostrou-se capaz de detetar várias faces na mesma imagem, apresentando alguns falsos positivos devido a variações de iluminação.

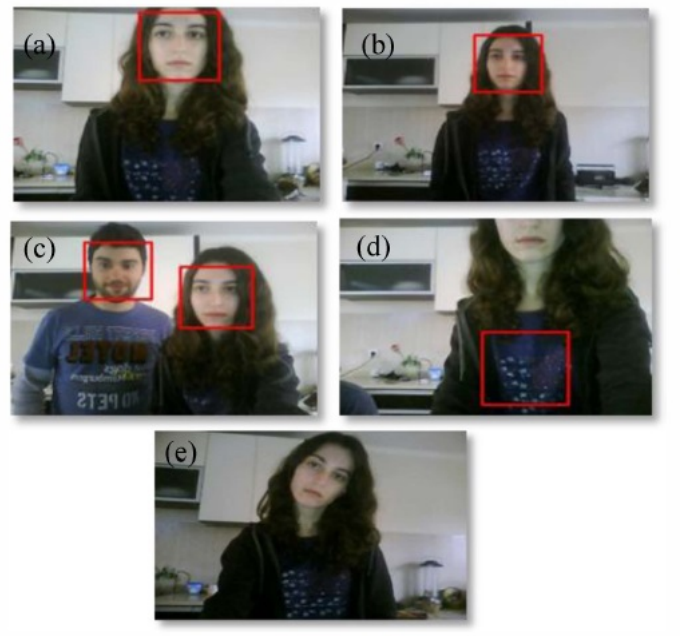


Figura 2.15: Resultados das experiências demonstrados no artigo de António Moreira e sua equipa [30].

2.2.5 *Histogram of Oriented Gradients*

O método de *Histogram of Oriented Gradients* (HOG) é muito utilizado em Visão por Computador pela sua capacidade de caraterizar objetos a partir do gradiente da imagem, de forma a obter informação capaz de ajudar a detetar que objetos estão presentes. Pelas suas caraterísticas de descrição genéricas este método é utilizado na deteção de muitos tipos de objetos tal como a deteção de pessoas em imagens [29].

A implementação deste método tem como início uma normalização da cor e filtragem da imagem para tornar mais eficaz o descritor produzido pelo gradiente. De seguida, passa para uma fase em que se divide a imagem em pequenos pedaços chamadas células, nas quais se cria um histograma de gradientes e orientação dos píxeis existentes na célula. A combinação destes histogramas representam um descritor da imagem. Na figura 2.16 está representado a ideia de bloco e célula utilizada pelo HOG.

Para o cálculo do gradiente é aplicada uma máscara para determinar as derivadas de primeira ordem tanto na horizontal como na vertical fazendo realçar as orlas e contornos da imagem. Após a obtenção do gradiente, é criado o histograma para cada célula. Cada píxel individual da célula contribui para um canal do histograma com o valor referente ao gradiente nesse píxel. O histograma é caraterizado por ter os valores distribuídos de 0^o a 180^o (figura 2.17) sendo chamado *"unsigned"* ou de 0^o a 360^o sendo chamado *"signed"*. Durante os testes foi descoberto que o histograma com a magnitude do gradiente *"unsigned"* e com o histograma dividido em 9 canais produzia melhores resultados para a deteção de pessoas em imagem [29].

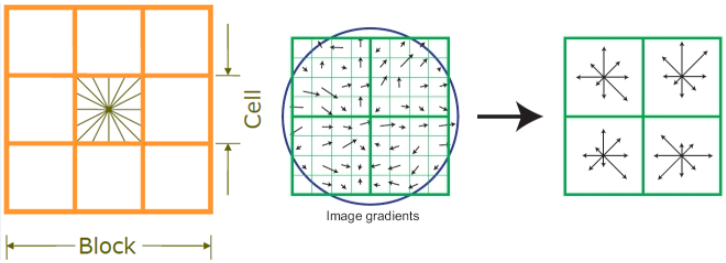


Figura 2.16: HOG célula e bloco, representação dos gradientes.

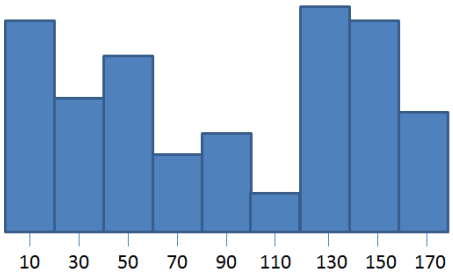


Figura 2.17: Uma das formas de representação do histograma dos gradientes na qual esta representada a amplitude em cada um dos 9 canais representativos de orientações de 0^o a 180^o (extraído de [29]).

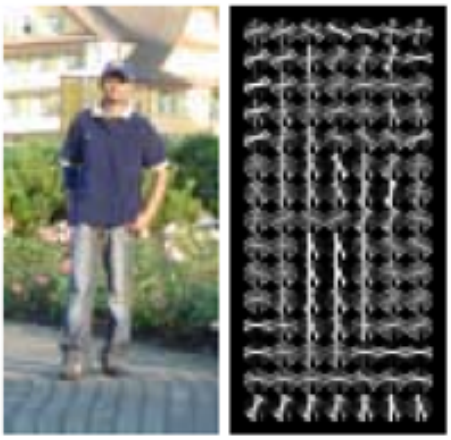


Figura 2.18: Representação de uma imagem de teste e do descritor com bloco em R-HOG (explicado mais a frente)(adaptado de [29]).

Para melhorar a deteção e combater as mudanças de iluminação e sombras, o histograma pode ser ajustado de forma a melhorar o contraste. Como valor de ajuste é usada uma intensidade calculada a partir da informação de uma região de píxeis maior do que a célula, chamada bloco. Este valor de ajuste é depois usado para normalizar todas as células que estão dentro do bloco. De forma a conseguires melhor resultados na normalização, os blocos podem ser sobrepostos para

que a mesma célula contribua para diferentes blocos, sendo importante uma boa normalização para obter bons resultados na deteção e classificação de objetos. O tamanho das células, blocos e a sua sobreposição são decididos conforme a aplicação e velocidade de cálculo desejada. A sobreposição é um parâmetro muito importante e que apresenta diferença no resultado final, pois como é referido em [29], a escolha de sobreposição de 3/4 apresentou uma melhoria na ordem de 4% em relação a testes sem sobreposição, para um dos casos estudados e apresentado no artigo. Após a normalização de todos os histogramas da imagem, é gerado um vetor que contém todas as células normalizadas de todos os blocos da imagem a processar.

Como forma geométrica dos blocos foram apresentadas duas soluções, R-HOG (blocos com grelhas em forma de quadrados) e C-HOG (blocos com a forma circular). Os blocos em formato R-HOG são caraterizados por três parâmetros, sendo eles o número de células por bloco, o número de píxeis por célula e o número de canais do histograma de cada célula. Os autores verificaram que para o efeito de detetar pessoas o melhor tamanho para as células é de 6x6 e que o melhor tamanho para blocos é de 3x3, sendo também de referir o número de canais do histograma que deve ser 9. Na figura 2.19 ilustra a comparação do desempenho para tamanhos diferentes de células e blocos.

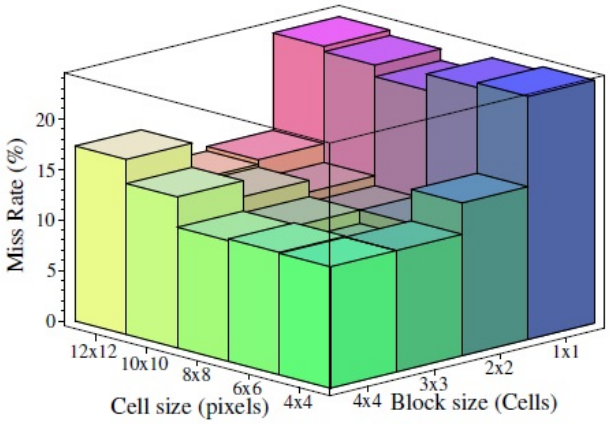


Figura 2.19: Comparação do desempenho de tamanhos diferentes de blocos e células e a percentagem de falsos positivos detetados por HOG testados por Dalal e Triggs (adaptado de [29]).

Os blocos em formato C-HOG são caraterizados por 4 parâmetros sendo eles o número de áreas angulares e raios, o raio da área central em píxeis e o fator de expansão para os raios das áreas radiais adicionais. Os blocos em C-HOG podem ser usados em dois formatos, um em que a célula central é apenas uma e outro em que a célula central é constituída por várias células, figura 2.20.

Para a normalização do descritor foram apresentadas quatro formas de cálculo. Nas experiências referidas em [29] chegaram à conclusão que três das equações apresentam resultados semelhantes, sendo difícil dizer qual a melhor. A outra equação apresenta piores resultados entre as quatro equações. De referir que todas as equações apresentam melhores resultados que o descritor sem estar normalizado.

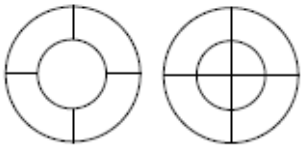


Figura 2.20: Representação das possíveis configurações dos blocos C-HOG. À direita o bloco C-HOG com apenas uma célula no centro e à esquerda com varias células no centro em áreas radiais (adaptado de [29]).

2.2.6 *Local Binary Patterns*

Local Binary Patterns (LBP) é um descritor de imagem criado para descrever texturas, tendo sido proposto em [31]. Este descritor é usado em vários campos de visão por computador, com realce para a deteção de caras, reconhecimento de faces e reconhecimento de expressões faciais, bem como na deteção de pessoas, tal como é demonstrado em [32].

O descritor LBP é aplicado em imagens na escala de cinzento e tem como principio a caracterização da imagem pela atribuição de um código binário a cada píxel, a partir da análise da vizinhança desse mesmo píxel. O código binário atribuído é obtido a partir da comparação do valor do píxel e dos píxeis da vizinhança obtendo o valor "0" se o píxel da vizinhança apresentar um valor maior ou igual ao píxel em análise, e o valor "1" caso contrário. Para o caso de LBP de 8 bits e raio igual a 1, a construção do número inicia-se no píxel que se apresenta em cima do píxel em análise e progride no sentido anti-horário e o número binário é criado do *bit* mais significativo para o menos significativo. Tendo sido obtidos todos os números binários de cada píxel da janela de análise, é criado o histograma que apresenta a quantidade de cada número binário presentes na janela a analisar. O histograma é por fim normalizado por normas semelhantes às usadas no método HOG [31, 32]. Em [33] é apresentado um sistema de deteção de pessoas com algum nível de ocultação utilizando HOG e LBP em conjunto. Na figura 2.21 está um exemplo do funcionamento do LBP. Criação do código binário a partir da análise dos píxeis em torno de um píxel para construção da sua representação binária e por fim um histograma de valores binários presentes.

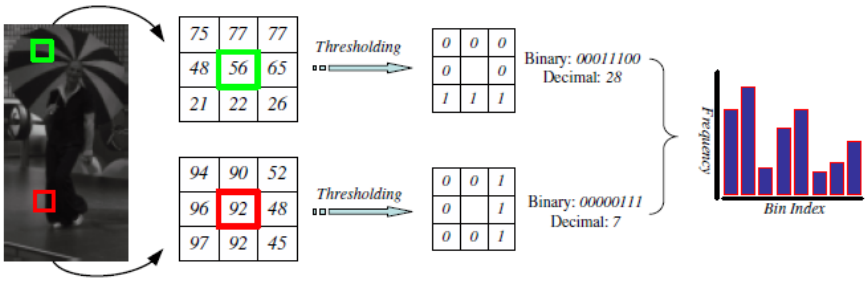


Figura 2.21: Imagem retirada do artigo [32].

2.3 Bibliotecas de Software

2.3.1 *OpenKinect*

O *OpenKinect* é um projeto *open source* criado com o objetivo de desenvolver aplicações, recorrendo ao equipamento *XBox Kinect* e fornecendo uma API para controlar os diferentes componentes do *Kinect* a partir de um PC ou outros equipamentos [34]. Este projeto é disponibilizado sem custos para quem o quiser usar e dispõe de bibliotecas para *Windows*, *Mac OS* ou *Linux*. Estas bibliotecas são desenvolvidas em torno de uma outra biblioteca de utilização do *Kinect*, a *libfreenect*. É possível utilizar o *OpenKinect* em várias linguagens de programação, com destaque para *C++*, *C#*, *Python*, *Java*, *Javascript*.

2.3.2 *OpenNI*

O *OpenNI* é um *Software Development Kit* (SDK) *open source* criado em novembro de 2010, para o desenvolvimento de aplicações ou bibliotecas com o uso de sensores 3D tais como o *Kinect* para interação com o computador apenas com o uso de movimentos [35]. Foi criado para interagir com sistemas de RGB-D, tais como o *Kinect* e o *Xtion PRO* e pode ser usado em várias linguagens de programação tais como *C++* e *C#*. Atualmente o *OpenNI* é constituído pela *Primesense* e uma vasta comunidade *online* que ajudam na evolução deste SDK.

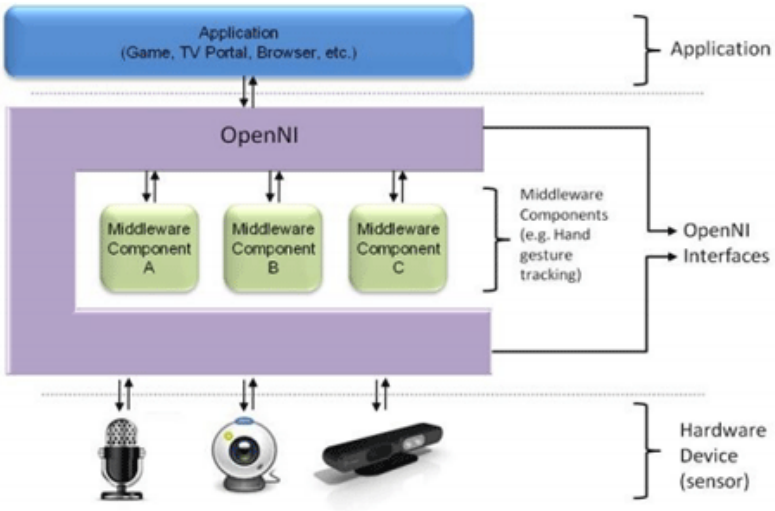


Figura 2.22: Arquitetura da API OpenNI (extraída de [35]).

2.3.3 *OpenCV*

OpenCV é uma biblioteca livre quer para uso académico quer comercial [36]. Foi concebida para ser usada em Visão Computacional tentando ser o mais eficiente e rápida possível, de forma a ser usada em aplicações de tempo real. Tem a possibilidade de ser usada em vários Sistemas

Operativos como *Windows*, *Linux*, *Android* entre outros, e em várias linguagens de programação tais como *C*, *C++*, *Python* e *Java*. Foi desenvolvida e otimizada em *C/C++*, com a capacidade de tirar partido de sistemas *multi-core*. Recorre ao uso de *Open Computing Language* (OpenCL) [37] que lhe oferece a capacidade de tirar partido de várias partes do hardware de um PC tal como o CPU e o GPU de forma a acelerar o processamento de imagem. É muito utilizado em todo o mundo em várias aplicações, desde arte interativa até à robótica.

2.3.4 *Kinect for Windows SDK*

Este SDK apresenta mais capacidades que os SDK referidos anteriormente, com a desvantagem de apenas ser possível ser usado por sistemas operativos *Windows* [10]. Entre as melhorias, é de realçar o acesso direto à informação dos sensores do equipamento e da garantia de qualidade da API e documentação, pelo facto de ter sido desenvolvido pelo proprietário e dedicado ao *Kinect*. Apresenta a capacidade de ser utilizado em várias linguagens de programação tais como *C++*, *C#* e *Visual Basic*.

O *Kinect for Windows SDK* foi apresentado em junho de 2011 para *Windows 7*, sendo desenvolvido pela *Microsoft* e lançado juntamente com a segunda versão do equipamento *Kinect*. Apesar de ter sido desenvolvido pela *Microsoft*, este SDK foi o último a aparecer tendo sido precedido pelo *OpenKinect* e o *OpenNI*.

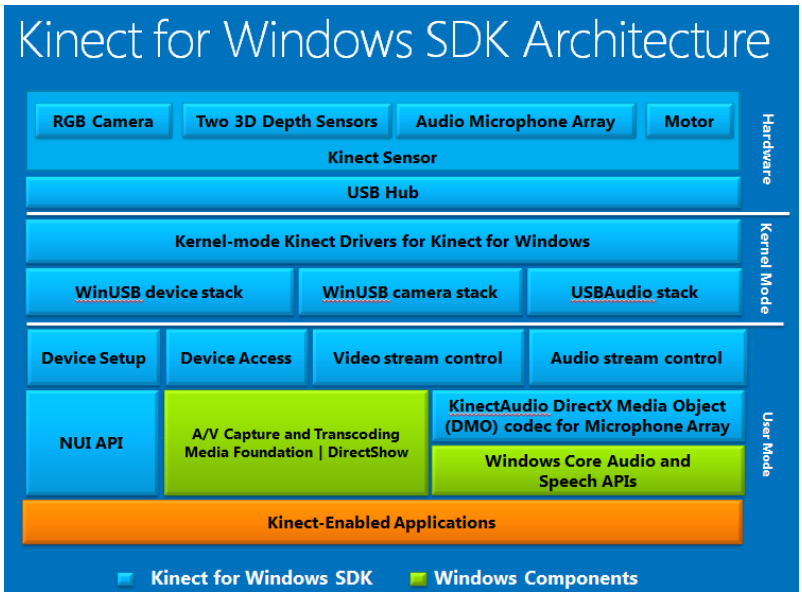


Figura 2.23: Arquitetura da API *Kinect for Windows* SDK

2.3.5 *Point Cloud Library*

Point Cloud Library (PCL) é um projeto aberto e de grande escala para processamento de imagem em nuvem de pontos em 2D ou 3D. Esta biblioteca é de uso livre e contém muitos algoritmos para filtragem, estimação de caraterísticas, reconstituição de superfícies e segmentação entre outros [38]. Pode ser usada em várias aplicações, como por exemplo, para filtrar ruído do contorno de imagens, segmentar partes relevantes da imagem, extrair pontos chave para criar um descritor e reconhecer objetos baseados na sua forma geométrica e criar superfícies a partir de pontos da imagem e visualizá-las.

É possível usar PCL em vários sistemas operativos tais como *Windows*, *Linux* entre outros. De forma a que o desenvolvimento desta biblioteca seja mais simples e potencialize a utilização em computadores de poder de computação reduzida e com pouco espaço, o PCL foi dividido em várias bibliotecas mais pequenas que podem ser compiladas em separado [39].

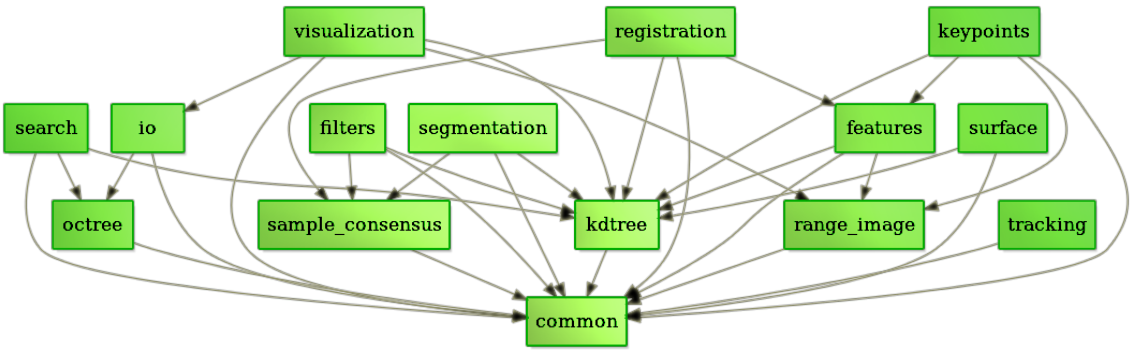


Figura 2.24: Arquitetura da biblioteca PCL

Capítulo 3

Plataforma Experimental

Na secção 3.1 deste capítulo é apresentado o equipamento utilizado no decorrer deste trabalho. Na segunda secção 3.2 é apresentado o programa criado para a captura das sequências de imagens, tal como algumas das sequências capturadas, as suas características e os requisitos do sistema. Por fim, na ultima secção 3.3, é apresentada a métrica a ser usada para classificar numericamente o desempenho do algoritmo criado.

3.1 Equipamento

Como referido em 2.1.1 o *Kinect for Windows*, que é uma adaptação do *Kinect for Xbox360* otimizada para o uso no PC, é uma câmara desenvolvida pela *Microsoft* e pela *PrimeSence*, tendo como principal caraterística a capacidade de capturar imagens RGB-D.

O *Kinect for Windows* é constituído por uma câmara RGB convencional, um emissor de infra-vermelho, uma câmara infra-vermelho, 4 microfones, motor e um acelerómetro como demonstrado na figura 3.1.

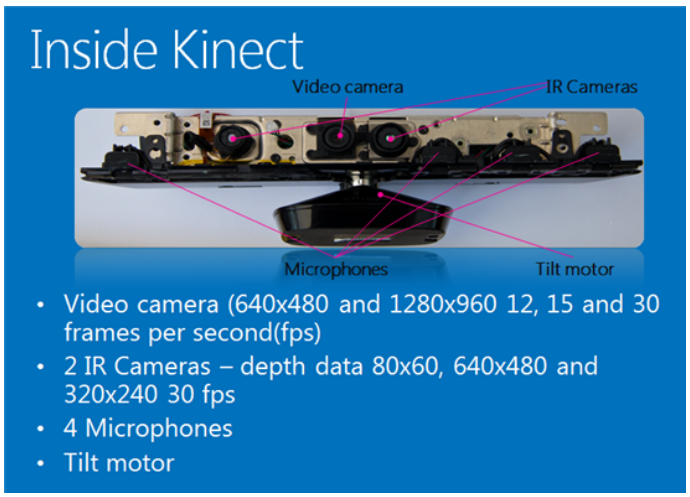


Figura 3.1: Disposição interna dos sensores do *Kinect for Windows*.

A câmara RGB convencional consegue produzir imagens em vários tamanhos tais como 640 x 480 e 1280 x 960 e efetuar capturas a 30 imagens por segundo (*Frames Per Secound* - FPS) [11]. O conjunto composto pela câmara e emissor infra-vermelho consegue obter imagens também a 30 FPS e com dimensões de 320x240 e 640x480. Os microfones são capazes de captar som com 24 bits de resolução na conversão analógico-digital e possuem a capacidade de eliminação de eco e supressão de ruído através de processamento interno. O motor e o acelerómetro permitem inclinar o *Kinect* entre +27º e -27º em relação ao nível normal da câmara, servindo o acelerómetro para estimar a sua posição angular e determinar se o *Kinect* está parado.

A câmara RGB e a câmara infra-vermelho possuem ângulos de visão diferentes, tendo a câmara RGB ângulos maiores. Desta forma, a câmara RGB produz imagens com mais informação da cena do que a câmara infra-vermelho. Esta diferença produz imagens ligeiramente desalinhadas entre si devido à diferença dos ângulos (como mostra a tabela 3.1) e ao facto das câmaras estarem uma ao lado da outra, sendo impossível captar imagens a partir do mesmo ponto.

Tabela 3.1: Ângulos de captura do *Kinect*, retirado de [40].

	RGB	Profundidade
HORIZONTAL	62º	58.5º
VERTICAL	48.6º	45.6º

Este equipamento possui dois modos de distância, o *default range* e o *near range*. As gamas de valores para os dois modos estão representadas na imagem seguinte (figura 3.2) retirada de [41], onde estão representadas também as distâncias descritas pela *Microsoft* que a API retorna.

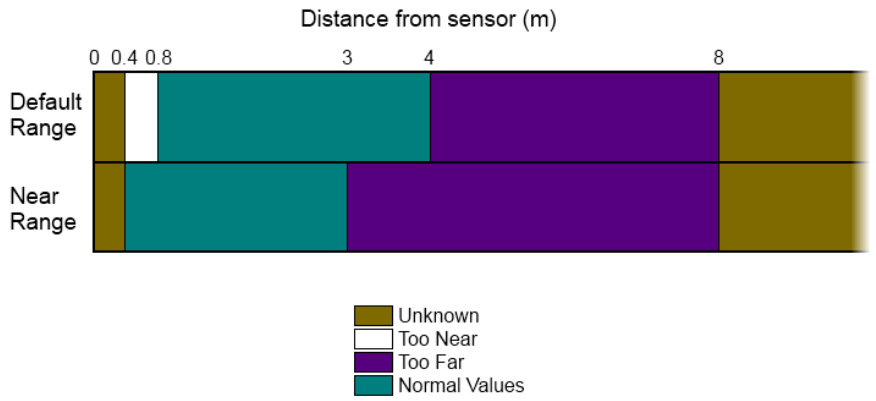


Figura 3.2: Distância descrita pela *Microsoft* [41].

Apesar da distância máxima com valores plausíveis apresentada na imagem ser de 8 metros, com o estudo do código de exemplos e pelos testes e capturas efetuadas, verificou-se que o *Kinect* consegue retornar distâncias até 16 metros utilizando toda a informação do vetor de distância proveniente do *Kinect*.

Para a obtenção das distâncias utilizando o emissor e a câmara infra-vermelho, o *Kinect* emite um padrão de pontos (figura 3.3) com o emissor infra-vermelho, que por sua vez são refletidos pela superfície dos objetos. A câmara infra-vermelho capta a imagem desses pontos e calcula o valor de distância para os píxeis da imagem (figura 3.5).

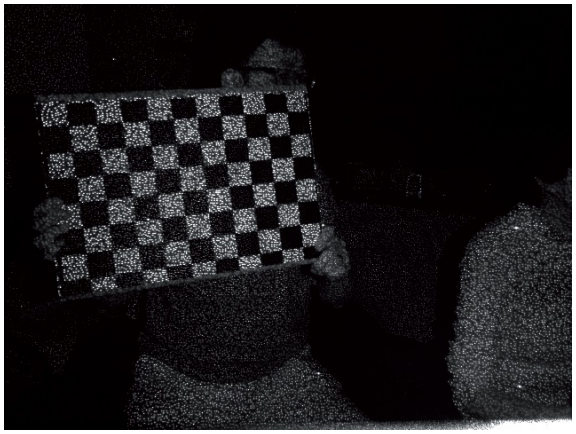


Figura 3.3: Exemplo dos pontos de infra-vermelho. A imagem apresenta a malha de pontos projetados pelo *Kinect*, imagem retirada de [42].

Este processo é repetido sempre que se captura uma nova imagem, retornando um vetor de valores de distância em milímetros para cada píxel. Os valores de distância não são em relação ao centro da câmara e ao objeto em frente dela, mas sim entre o plano em que está o *Kinect* e o objeto, como indica a figura 3.4.

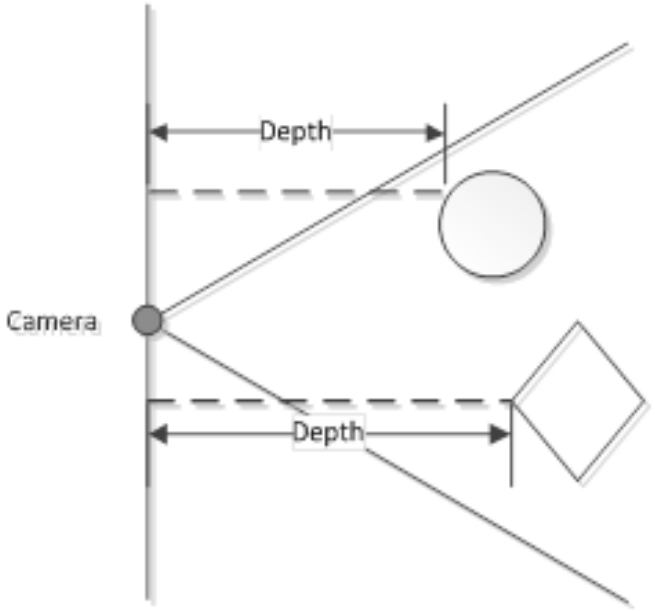


Figura 3.4: Ponto de medição, imagem retirada de [41].

Figura 3.5: Diagrama do PrimeSensorDepth, imagem retirada de [43].

3.2 Capturas

Foi desenvolvido um código que efetua a inicialização do *Kinect* e captura de imagens por parte do mesmo. O *Kinect* tem a capacidade de capturar vários dados, embora na nossa aplicação só tenham sido utilizadas a imagem RGB e a imagem de profundidade.
Foram estudados códigos exemplo, que estão escritas em *C#*, *C++* e *Visual Basic*, com várias funcionalidades desde o uso apenas da câmara RGB ou profundidade até ao uso de todos os sensores disponíveis do *Kinect* em conjunto. Para potenciar a utilização de um conjunto de diversas ferramentas, incluindo o OpenCV, o programa criado foi escrito em *C++*.

Para obter um melhor alinhamento e aumentar a correlação entre as imagens RGB e as de profundidade o programa criado faz uma captura das imagens, se ambas forem possíveis no mesmo ciclo. Desta forma existe uma maior correlação dos dados em ambas as imagens, pois são capturadas ao mesmo tempo, no entanto esta especificação do programa causa um impacto na taxa de capturas. Como o equipamento tem a capacidade de capturar imagens a 30 FPS, mas a captura de imagens RGB e de profundidade são assíncronas, a taxa de captura será sempre inferior a 30 FPS.

Após o estudo do código e de como obter as imagens do *Kinect* foi criada uma primeira versão do programa de captura que obtinha as imagens do *Kinect* com uma resolução de 640x480 e profundidades compreendidas entre 0.8m e 4m, sendo as profundidades apresentadas numa escala de cinzento.

Na figura 3.6 à esquerda vemos um exemplo da imagem de profundidade capturada em escala

de cinzento e na imagem da direita vemos a imagem RGB capturada no mesmo instante. Como podemos ver na figura 3.6, a profundidade de 4 metros é uma profundidade limitada para cenários de Ambiente Assistido e a escala de cinzento não é a melhor opção de representação. Por esses motivos foram estudadas e implementadas alterações ao código inicial de forma a obter maior profundidade.

Figura 3.6: Primeiras capturas com o programa criado para o *Kinect*.

Na figura 3.7 à esquerda vemos um exemplo da imagem de profundidade em escala de cinzento e na imagem da direita vemos a imagem RGB no mesmo instante. Neste par já podemos ver na imagem de profundidade valores referentes a distâncias superiores a 4 metros (neste caso a profundidade máxima registada é cerca de 10 metros).

Figura 3.7: Capturas com o programa criado para o *Kinect* em testes de distâncias.

Para melhorar a visualização das diferentes profundidade e facilitar a analise da imagem de profundidade foi criada uma função que recebe a imagem de profundidade e a representa utilizando uma palete de cores usando os 3 canais RGB, esta função é detalhada na secção 3.2.1.

Na figura 3.8 podemos ver a mesma ideia representada pelas imagens anteriores, mas utilizando uma palete de cores para representar as profundidades capturadas.

Figura 3.8: Capturas com o programa criado para o Kinect com utilização de cor para representar distâncias (distância máxima de 10 metros).

3.2.1 Palete de Cores

Para a conversão de profundidade em cor foi criada uma função que recebe um valor de profundidade e retorna o valor dos 3 canais de cor para essa distância. Esta função foi criada a partir do estudo da representação das cores em RGB. Na figura 3.9 está apresentado a evolução dos valores do canal R, G e B e associado a essa evolução esta os valores de profundidade a representar. No eixo das abcissas temos a representação da profundidade, para efeitos gerais ela é representada em intervalos aqui designados de "saltos". O intervalo é calculado dividindo a distância máxima admissível em 6 partes iguais. No eixo das ordenadas temos o valor de 0 a 255 que o canal deve ter para representar a distância.

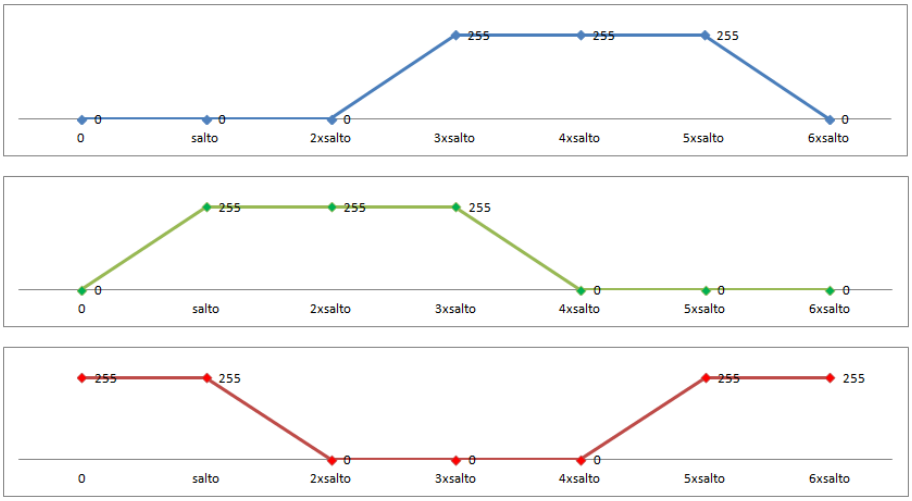


Figura 3.9: Regras usadas para conversão de distância para cor.

Este sistema de conversão de distância foi desenvolvido no decorrer da dissertação com a intenção de usar toda a gama de cores do espetro. Foram também usadas as cores "branco" e "preto" para dar indicações, não de uma distância numérica, mas de problemas ou indicações sobre a medição no píxel. A cor branca foi usada para indicar píxeis para o qual não se consiga

obter uma resolução em distância indicando ruído ou problemas de medição por falta de deteção da luz infra-vermelho. A cor preta foi utilizada para indicar píxeis em que se consegue obter uma distância, mas esta está acima da distância máxima definida no programa.

3.2.2 Alinhamento da imagem

Devido aos ângulos de captura de imagens serem diferentes para a câmara RGB e a câmara de infra-vermelhos e estando elas lado a lado, existe uma diferença na informação capturada pelas câmaras e as imagens estão desalinhadas, o que dificulta a extração da distância de um píxel representado na imagem RGB, pois na imagem de profundidade o píxel correspondente não se encontra nas mesmas coordenadas. A figura 3.12 ilustra em parte a relação entre os diferentes ângulos.

Para tentar diminuir o efeito do desalinhamento das câmaras, foi criada uma função que recebe a imagem RGB e altera a mesma de forma a ficar alinhada com a imagem de profundidade. Tem de ser a imagem de RGB a ser alterada pois é esta que possui mais informação, sendo esta a ser recortada de forma a ser alinhada com a imagem de profundidade.

Na figura 3.10 podemos ver pintado a azul os contornos da imagem RGB pelo método Canny e a vermelho os contornos da imagem de profundidade pelo mesmo método.

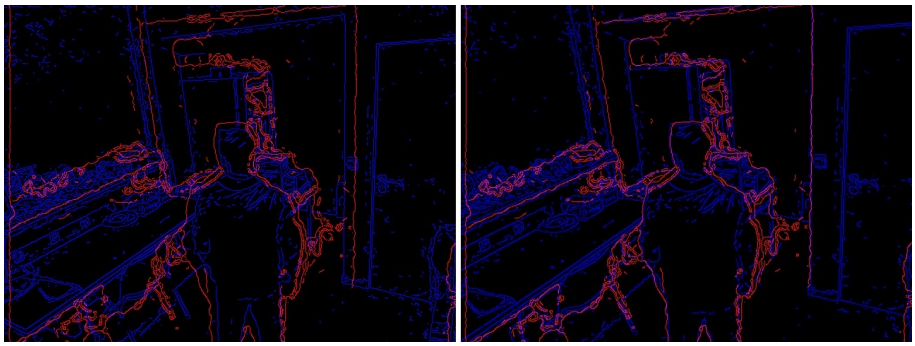


Figura 3.10: Exemplo de alinhamento das imagens. Imagem da esquerda representa as imagens antes do alinhamento e a imagem da direita as imagens depois do alinhamento. Os contornos da imagem RGB estão representados a azul e os contornos da imagem de profundidade estão representados a vermelho.

Na imagem da esquerda está representada a sobreposição das imagens sem a função de alinhamento, onde podemos ver que as linhas azuis e vermelhas não coincidem, ao contrário do que aconteceria se as imagens estivessem alinhadas. Na imagem da direita está o resultado da imagem após o alinhamento, onde podemos ver que as linhas azuis e vermelhas estão praticamente sobrepostas e que as linhas vermelhas aparentam ser um segundo contorno para as linhas azuis.

Uma técnica testada para o alinhamento das imagens, foi o uso dos cantos dos objetos presentes nas imagens. Esta técnica quando foi estudada e pensado o seu modo de funcionamento, tinha como especial característica ser automática, utilizando os cantos presentes em ambas as imagens

de profundidade e de RGB para determinar uma matriz de transformação que alinhava as imagens. Esta ideia apresentou alguns problemas devido ao ruído da imagem de profundidade, que apresenta cantos falsos e não estáveis no espaço ou seja, devido ao ruído o valor da sua posição não é constante. Alguns desses cantos falsos existem devido ao efeito de sombra causado pelo infra-vermelho do *Kinect*, reflexão do infra-vermelho e pela absorção do infra-vermelho por parte de alguns objetos que criam manchas de ruído na imagem, sendo admitidos cantos que não existem na realidade.

Outro problema detetado foi o elevado número de cantos da imagem RGB em relação à imagem de profundidade e à diferença de posição entre eles, o que dificulta a correlação entre pontos para o alinhamento como ilustrado na figura 3.11.

Figura 3.11: Detetor de cantos. Na imagem da esquerda podemos ver representados os cantos detetados na imagem RGB e na imagem da direita os cantos detetados na imagem de profundidade.

Outra técnica estudada e por fim utilizada na dissertação foi desenvolvida a partir do estudo dos ângulos das câmaras. Com o estudo dos ângulos de captura das câmaras, foi determinada a área da imagem RGB para a qual não existe informação de profundidade. Como o centro de ambas as imagens é aproximadamente o mesmo, foi calculado (equações 3.1 e 3.2) o ponto na imagem de RGB a qual corresponde ao canto superior esquerdo da imagem de profundidade. Uma vez calculados estes valores, é cortada à imagem RGB a área para a qual não existe informação de profundidade. Para fazer a sobreposição das imagens estas têm de ser, preferencialmente, da mesma dimensão. A sub-imagem RGB é redimensionada de forma a ficar com as mesmas dimensões da imagem de profundidade. Apenas com esta sobreposição não se obtém um alinhamento aceitável das imagens. Dessa forma é necessário fazer um ajuste aos valores obtidos pelo cálculo. O ajuste é efetuado para corrigir e combater o erro da imagem de profundidade de forma a obter melhor alinhamento. O ajuste é efetuado com testes empíricos para obter os valores de ajuste. São desenhados os contornos das imagens pelo método de Canny e efetuada uma sobreposição dos contornos de ambas as imagens como ferramenta visual, para efetuar os ajustes manuais e verificar o alinhamento das imagens, como demonstrado na figura 3.10. No algoritmo 1 é apresentado o funcionamento da função de alinhamento que recebe os parâmetros determinados pelo cálculo e os valores de ajuste para recortar a imagem RGB e redimensionar a mesma.

Algorithm 1 Alinhamento

1: **função** ALIGNFRAME(left, top, offsetleft, offsettop)
2: w = numero de colunas da imagem - left*2;
3: h = numero de linhas da imagem - top*2;
4: Imagem de saída = imagem de entrada recortada por *pixel* (left + offsetleft, top + offsettop) com w colunas e h linhas
5: Redimensionar a imagem de saída cortada para dimensão da imagem de profundidade
devolve Imagem de saída
6: **fim função**

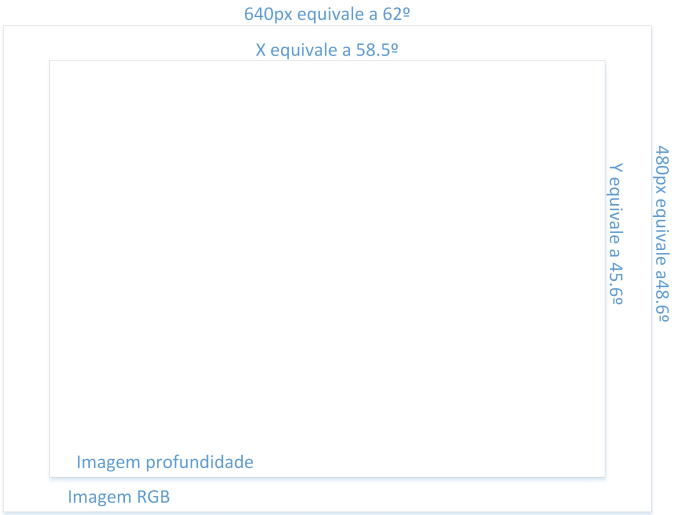


Figura 3.12: Relação entre os ângulos e dimensões.

$$x\ ponto\ superior\ esquerdo = 640 - int(\frac{58.5^o \times 640}{62^o}) \tag{3.1}$$

$$y\ ponto\ superior\ esquerdo = 480 - int(\frac{45.6^o \times 480}{48.6^o}) \tag{3.2}$$

3.2.3 Infra-vermelho

Como o *Kinect* efetua a medição da profundidade a partir de um mapa de disparidade gerado por infra-vermelho, foram efetuadas algumas capturas para testar interferências da luz do sol e da luz ambiente ou artificial nas imagens de profundidade. Na primeira fase esteve em análise a interferência de luz artificial proveniente da iluminação da sala.

Na figura 3.13 podemos ver o efeito da luz direta proveniente de uma luz artificial no teto da sala e na figura 3.14 podemos ver o efeito da luz semidireta. Nas imagens podemos ver píxeis a branco que representam píxeis para os quais não se consegue obter um valor de distância devido a

ruído ou interferências. Na figura 3.15 podemos ver um teste de reflexão da luz artificial no chão, que parece ter apenas uma influência mínima.

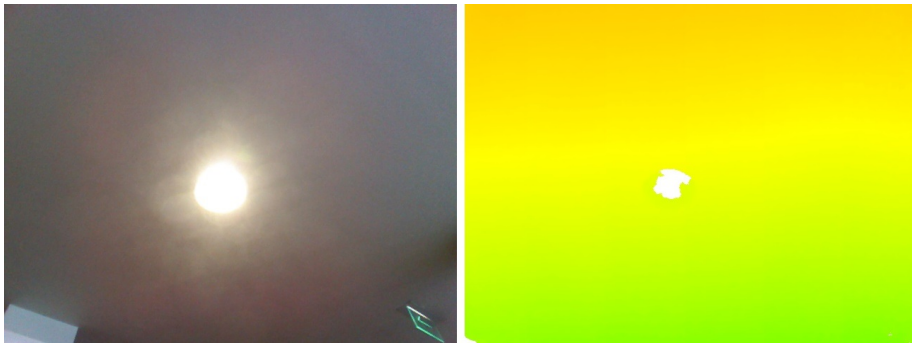


Figura 3.13: Luz artificial direta para o *Kinect*.

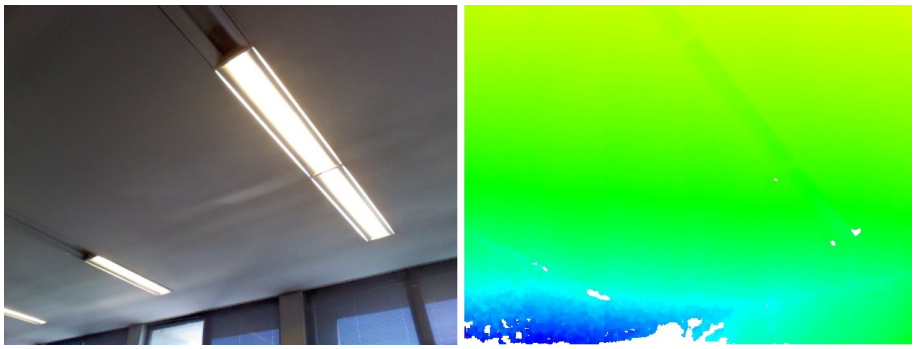


Figura 3.14: Luz artificial semidireta para o *Kinect*.

Figura 3.15: Reflexão de luz artificial no chão

Também foi analisado o impacto da luz do sol de alguma forma nas medições do *Kinect* captando imagens com luz do sol direta. Foi realizada uma experiência que consistiu em tapar a parte responsável por emitir o infra-vermelho por parte do *Kinect* e efetuar capturas em vários ângulos e posições para verificar se o *Kinect* retornava alguma distância dentro da gama de funcionamento.

Na figura 3.16 podemos ver um dos resultados da experiência acima descrita. Este resultado foi obtido apontando o *Kinect* para uma das janelas viradas para o sol. De realçar que no decorrer da experiência foi detetado que este género de resultados não acontece em todas as capturas. O *Kinect* foi movimentado de forma a estar com diferentes ângulos e posições em relação à janela e verificou-se que apenas em algumas posições semelhantes à da figura é que se verificava o retorno de valores de distância.

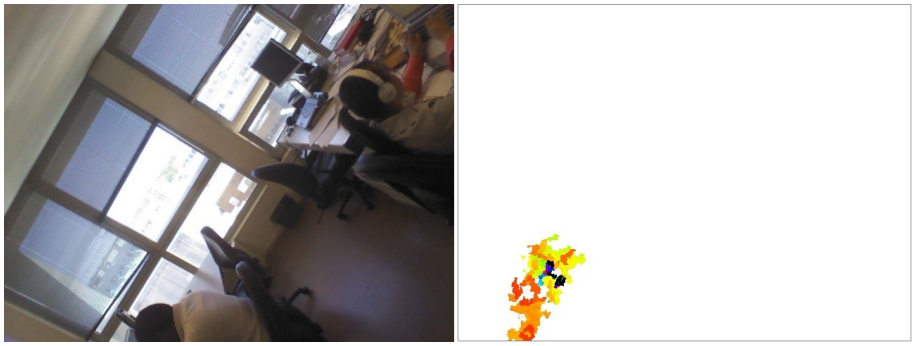


Figura 3.16: Teste do infra-vermelho na janela com sol.

Ainda dentro da análise de possíveis situações de interferência no infra-vermelho do *Kinect*, foi efetuado um novo teste que consistia em capturas de imagens através de vidros, pretendia-se verificar se o *Kinect* era capaz de obter distância de objetos através do vidro ou se o infra-vermelho sofria distorção ou reflexão através do vidro.

Na figura 3.17 podemos ver 3 capturas de imagens nas quais conseguimos obter valores de distância para os objetos dentro do móvel através do vidro, indicando que é possível obter distâncias de objetos posicionados atrás de vidro.

Ao longo das várias capturas efetuadas com o objetivo de testar as imagens infra-vermelho, o equipamento e o programa, foram detetadas algumas situações de ruído a distâncias curtas inicialmente não previstas. Após uma análise das imagens, concluiu-se que o ruído tem 3 fontes principais: ruído devido a corpos negros; reflexão da luz infra-vermelho e efeito sombra.

Nas imagens 3.18 e 3.19 podemos ver o ruído provocado por corpos negros. Como se verifica em ambas as imagens, existem corpos negros que apresentam píxeis brancos na imagem de distância, ou seja, não retornam distância para esses corpos. Este ruído é causado pela absorção da luz infra-vermelho por parte desses corpos. Podemos ver que na figura 3.18 o bloco preto mais em baixo não apresenta por completo valores de distância, bem como o cabelo preto da pessoa presente na imagem não apresenta valores, ou seja, aparece uma mancha branca na imagem de profundidade. Na imagem 3.19 podemos ver que o saco do caixote do lixo não apresenta valores de distância devido à sua cor.

Outra das fontes de ruído detetada é a reflexão da luz infra-vermelho em planos brilhantes e inclinados. Esta reflexão provoca "buracos" nas medições de distância, pois o feixe infra-vermelho responsável por indicar a distância de um conjunto de píxeis não foi refletido na direção do *Kinect*, fazendo com que este não consiga efetuar a medição. Este fenómeno pode ser observado

nas figuras 3.7, 3.8 e 3.19, nas quais podemos ver que em algumas áreas do chão e superfícies, principalmente o mais longe da câmara ou com grande inclinação em relação ao *Kinect*, não existe valor de distância para essas áreas.

O terceiro motivo de ruído encontrado é causado pela projeção da malha de infra-vermelho para a produção do mapa de disparidade. Como esta malha é projetada em forma de leque a partir de um ponto, existe a criação de uma sombra para objetos muito perto do *Kinect*.

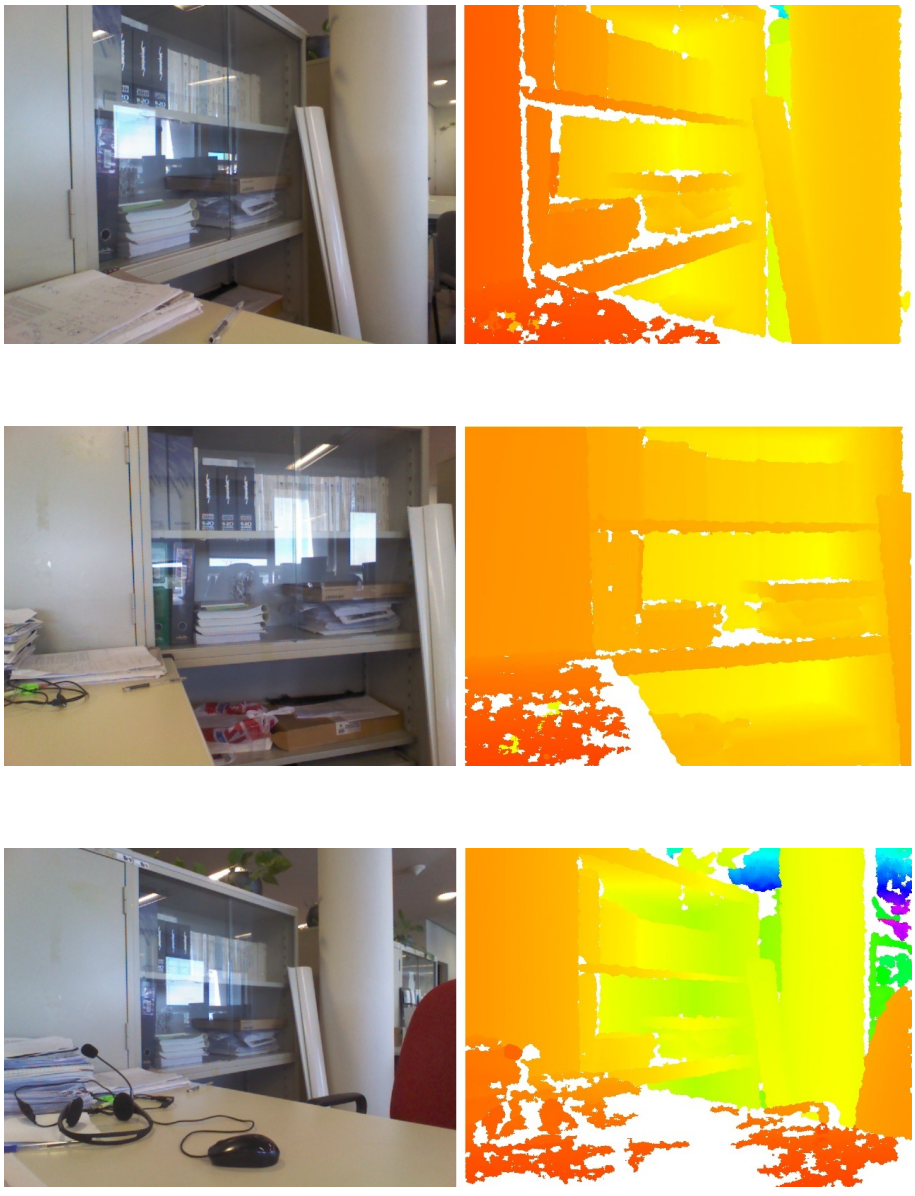


Figura 3.17: Medição através vidro.

Figura 3.18: Interferência devido a cor. Rodeado a vermelho podemos ver as áreas que apresentam ruído devido à cor, neste caso o cabelo da pessoa e a mesa mais perto da câmara.

Figura 3.19: Interferência devido a cor. Rodeado a vermelho podemos ver uma área de ruído devido a cor negra do saco do lixo presente na imagem.

Na figura 3.20 pode-se ver o efeito sombra produzido pelo feixe infra-vermelho do *Kinect*. Na imagem da direita pode-se ver uma mancha branca do lado direito da pessoa com o contorno muito semelhante à silhueta da pessoa. Essa silhueta é produzida por falta de valores de distância causados pelo efeito sombra da abertura do feixe de infra-vermelho.

Como o *Kinect* utiliza infra-vermelho para efetuar medições e a variação de luz no cenário pode ter influência na medição, foi realizado capturas com diferentes condições de iluminação para analisar o impacto na imagem de profundidade. Na figura 3.21 podemos ver duas capturas com condições de iluminações diferentes causadas pelo abrir e fechar dos estores das janelas. Com a análise destas imagens pode-se concluir que os valores medidos pelo infra-vermelho em distâncias grandes (neste caso cerca de 10 a 12 metros) são um pouco influenciados pela iluminação da sala. Podemos constatar que a quantidade de ruído no fundo da sala é maior na imagem com maior iluminação. Apesar dessa diferença nas medições devido à iluminação conclui-se que a imagem de distância do *Kinect* apresenta elevada robustez à variação de iluminação comparando com a mesma situação nas imagens RGB.

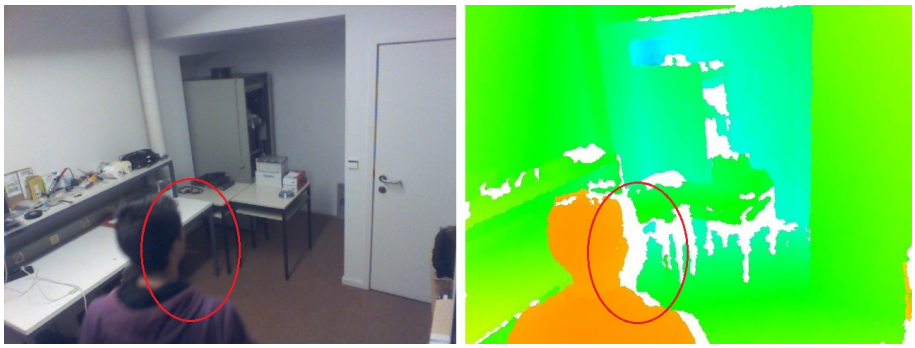


Figura 3.20: Produção de sombra por parte da projeção do infra-vermelho. Rodeado a vermelho está realçada uma das zonas que apresenta o efeito sombra na imagem de distância.

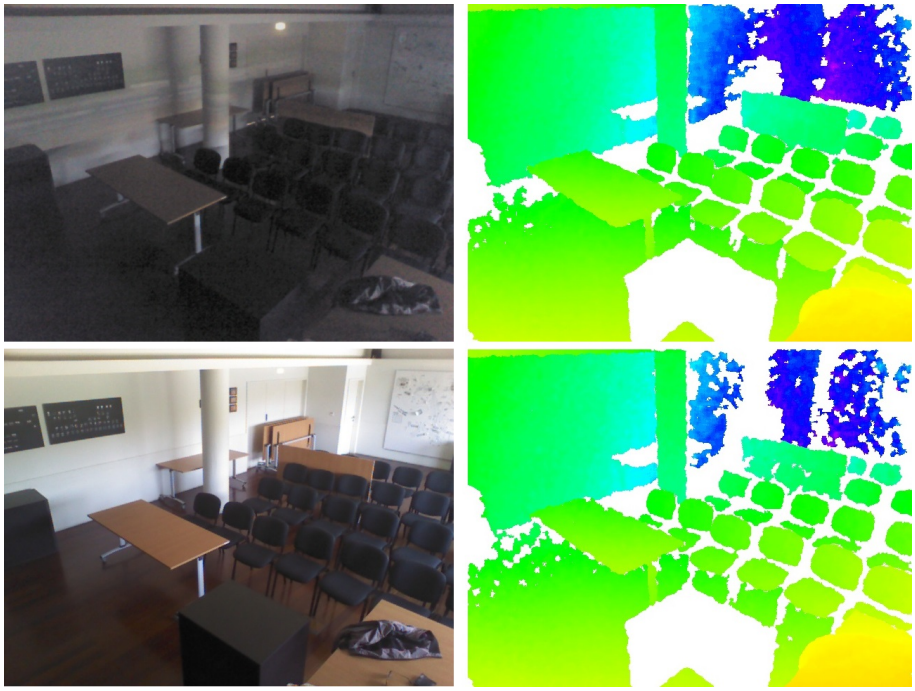


Figura 3.21: Comparação de imagem com muita luz e pouca luz.

3.2.4 Filtragem

De forma a tentar melhorar as imagens de profundidade e eliminar ou diminuir o ruído da imagem, produzido por falhas de medição, foi estudado e desenvolvido um algoritmo de filtragem. A ideia base deste algoritmo é de preencher os "buracos" criados por erro nas medições através do cálculo do valor que ele devia tomar a partir dos valores que se encontravam ao seu redor a uma determinada distância. A figura 3.22 apresenta um exemplo da aplicação do filtro à imagem da esquerda sendo o resultado a imagem da direita.

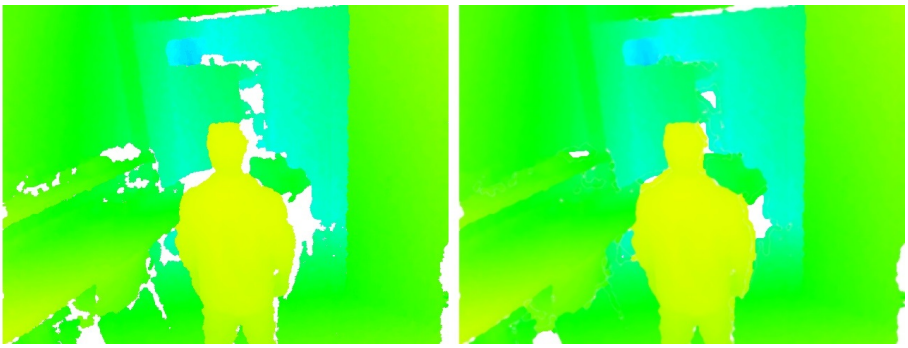


Figura 3.22: Exemplo de resultado da filtragem, imagem da esquerda antes do filtro e a imagem da direita resultado.

O Algoritmo da Filtragem (algoritmo 2) começa por percorrer todos os píxeis da imagem a procura de um píxel com cor branca. Os píxeis de cor branca representam píxeis sem valor de distância associado. Uma vez encontrado um píxel de cor branca, são percorridos todos os píxeis em torno desse píxel que se encontrem dentro de uma matriz, com dimensões atribuídas por experimentação, com centro no píxel branco encontrado. Ao percorrer os píxeis dentro da matriz descrita, é determinado o valor de menor profundidade e atribuída a sua cor ao píxel branco.

Este algoritmo foi testado numa sequência de imagens na qual o ator não apresenta grande nível de ocultação. Neste teste observou-se bons resultados na redução de ruído visto que o ruído presente correspondia maioritariamente a objetos com menor distância do que os elementos circundantes. Quando foi testada esta filtragem com o algoritmo de subtração de fundo implementado e com um cenário com maior nível de ocultações, verificou-se que introduzia um pequeno ruído na subtração não sendo por isso utilizada no algoritmo final. No algoritmo 2 a função $profundidade()$ recebe um píxel colorido e retorna a profundidade representada por essa cor e a função $Color()$ recebe uma profundidade e retorna a cor correspondente.

3.2.5 Sequências capturadas

Devido à falta de sequências de dados com as caraterísticas desejadas para as experiências a realizar, foram capturadas sequências para análise e teste do algoritmo criado. Estas continham um ou dois "atores" e diversas situações de ocultações. Para o funcionamento do algoritmo proposto e aumentar o desempenho do sistema o sistema de subtração de fundo (apresentado em 4.2) necessitar de um cenário sem pessoas ou objetos moveis no inicio da sequência e a posição do *Kinect* escolhida de forma a diminuir a percentagem de ocultação das pessoas. Por essa razão foram definidas algumas condições iniciais para a captura: do *Kinect*, deve estar num ponto elevado e direcionado para o centro da sala; no inicio da captura a sala tem de estar sem "atores", ou seja, a sala tem de estar sem pessoas e objetos com movimento.

Estas sequências foram realizadas em dois locais com diversos objetos de forma a criar algum nível de ocultação para teste do algoritmo. A primeira sala consiste numa sala de pequenas dimensões e com poucas ocultações imitando uma pequena sala de trabalho ou escritório. A segunda

sala onde foram efetuadas capturas é uma sala de maiores dimensões, mais ampla e com uma maior probabilidade de ocultações. Esta sala pode representar uma divisão de maiores dimensões dentro de uma casa, como por exemplo uma sala de estar ou de jantar.

Algorithm 2 Filtragem

```
    função FILTRAGEM( imagem de entrada)
2:      variavel Imagem de saida
        variavel val
4:      variavel img ← imagem de entrada
        variavel matrix dim ← 15
6:      variavel depth ← profundidade maxima
        enquanto todos os píxeis da imagem não forem percorridos faça
8:          se pixel de img == branco então
                enquanto todos os píxeis da matriz com dimensão (matrix dim) e centro no píxel
    branco de img faça
10:                 val ← profundidade(pixel)
                    se val <= depth então
12:                     depth ← val
                        pixel imagem saida ← Color(val)
14:                 fim se
                fim enquanto
16:         senão
                pixel imagem sada ← pixel de img
18:         fim se
        fim enquanto
         devolve Imagem de saida
20: fim função
```

Figura 3.23: Cenário 1 sem pessoas.

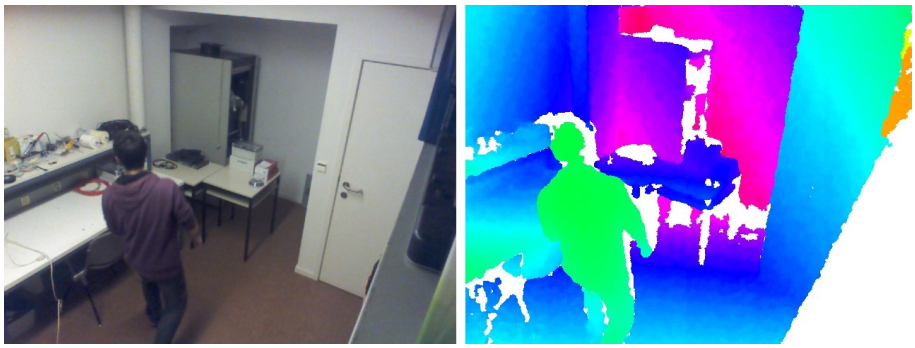


Figura 3.24: Cenário 1 com pessoas e sem ocultação.

Figura 3.25: Cenário 3 sem pessoas.

Figura 3.26: Cenário 3 com pessoas e ocultação.

3.3 Métricas de avaliação

3.3.1 Informação de Referência

Para a validação dos dados e análise dos mesmos é necessário comparar com a informação/*output* ideal. Este conjunto de dados chamados de *Ground Truth* são tipicamente gerados a partir de marcação manual das imagens. A marcação é efetuada por uma pessoa que analisa

visualmente cada imagem a usar e que marca manualmente a área da imagem que contém uma pessoa. Desta marcação é gerado um ficheiro que contém os valores de posição da caixa referente a área demarcada. O ficheiro gerado é um ficheiro xml que contem uma *tag* por *frame* com a informação das marcações da *frame* correspondente.

3.3.2 Métricas

Com o intuito de uma avaliação numérica dos resultados obtidos foram estudadas as métricas de avaliação do algoritmo [44, 45]. Foram implementadas e usadas as métricas baseadas nas métricas apresentadas em [45]. Neste sistema de métricas são retirados 3 parâmetros através das imagens usadas:

Positivo Verdadeiro (PV), número de deteções do algoritmo que coincidem com o *Ground Truth*.

Falso Negativo (FN), número de *Ground Truth* sem correspondência por parte dos detetores do algoritmo.

Falso Positivo (FP), número de deteções sem *Ground Truth* associado.

Nos parâmetros apresentados é considerado que a deteção do algoritmo coincide com o *Ground Truth* se o centro da caixa do detetor estiver dentro da caixa do *Ground Truth*. Para cada *Ground Truth* apenas pode existir uma caixa do detetor associado, ou seja, se existir uma caixa de *Ground Truth* e mais do que uma dos detetores, a caixa dos detetores mais próxima do *Ground Truth* é associada ao *Ground Truth* e as outras são consideras Falsos Positivos. Após a passagem de todas as imagens e retirados os 3 parâmetros referidos, são calculadas as métricas.

$$False\ Alarme\ Rate\ (FAR) = \frac{FP}{(PV + FP)} \tag{3.3}$$

$$Detection\ Rate(DR) = \frac{TP}{(PV + FN)} \tag{3.4}$$

$$Positive\ Prediction(PP) = \frac{PV}{(PV + FP)} \tag{3.5}$$

O *Positive Prediction* (PP) (3.5), representa a razão entre o número de deteções por parte do algoritmo com correspondência com o *Ground Truth* e a soma do número de correspondências e não correspondências por parte do algoritmo. Este valor representa a fiabilidade da informação, ou seja, quanto maior for este valor maior é a certeza da deteção. O *Detection Rate* (DR) (3.4), representa a razão entre o número de deteções positivas por parte do algoritmo e a soma de deteções positivas e o número de deteções negativas, ou seja, objetos do *Ground Truth* que o algoritmo não encontrou. Este valor representa a quantidade de vezes que o algoritmo afirma existirem pessoas em relação ao número de vezes que existem realmente pessoas. Quanto maior for este número

maior é a taxa de deteção do algoritmo, querendo dizer que o algoritmo deteta praticamente todas as pessoas existentes. O *False Alarme Rate* (FAR) (3.3), representa a razão entre o número de deteções por parte do algoritmo sem correspondência com o *Ground Truth* e a soma do número de correspondências e não correspondências por parte do algoritmo. Este valor apresenta semelhanças ao PP sendo que este é esperado ser o mais baixo possível, ou seja, quanto mais baixo o valor menos vezes o algoritmo indica a existência de pessoas quando na realidade não existem.

Capítulo 4

Deteção de Pessoas com Fusão de dados

Neste capítulo é apresentado um algoritmo para a deteção de pessoas com fusão de dados captados pelo *Kinect*. Na secção 4.1 é descrito a proposta apresentada e os seus constituintes. Na penúltima secção 4.2 é discutido e apresentado o sistema de Subtração de Fundo implementado. Na ultima secção 4.3 é descrito os detetores testados e usados na proposta. No fim deste capitulo é apresentado duas figuras que ilustram o funcionamento de duas subtrações de fundo apresentadas.

4.1 Algoritmo

O algoritmo proposto tem como principio básico a procura de regiões de interesse na imagem para ser efetuada a deteção de pessoas nessas regiões. As regiões de interesse são áreas da imagem em que existe movimento de uma pessoa, sendo este movimento detetado por subtração de fundo, como mostra o conceito do sistema na figura 4.1.

De início, e como já foi referido em 3.1, foi criado um programa para efetuar captura de imagens com o *Kinect*. Este programa grava imagens RGB e de profundidade provenientes do *Kinect*. As imagens de profundidade são coloridas (3.2.1) e representam profundidades até 16 metros.

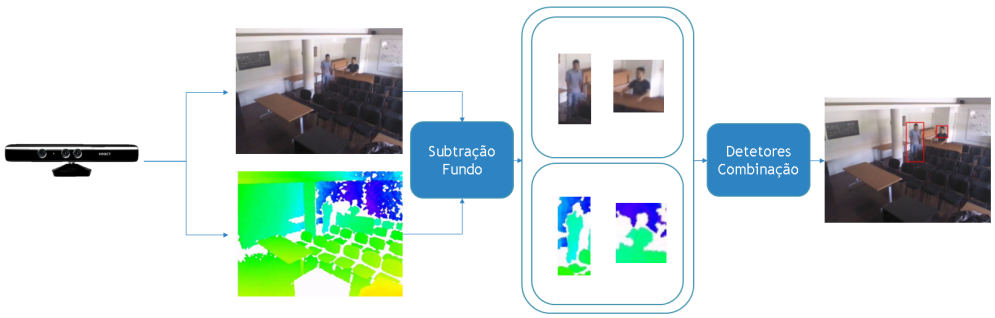


Figura 4.1: Conceito do sistema proposto.

Uma vez capturadas as imagens, estas são carregadas para um programa que aplica o algoritmo. O algoritmo começa por tratar as imagens. Neste tratamento a imagem RGB é alinhada com a imagem de profundidade utilizando o método descrito na secção 3.2.2 e a imagem de profundidade é filtrada de forma a minimizar a quantidade de "buracos" e ruído das medições. A filtragem efetuada na imagem de profundidade é feita de forma a tentar diminuir o ruído da imagem, através de uma dilatação seguida por uma erosão de núcleo mais pequeno que a dilatação, por fim um filtro de mediana para suavizar a imagem. O método de filtragem discutido em 3.2.4 não foi utilizado neste algoritmo porque na existência de algum objeto entre uma pessoa e a câmara produz uma correção de ruído na pessoa incorreta, induzindo em erro e diminuindo a área da pessoa.

Com as imagens pré-processadas é dado inicio ao processo de Subtração de Fundo descrito em 4.2.3. O processo de subtração gera uma imagem que indica áreas que apresentam objetos diferentes do fundo. Estas áreas são então cortadas da imagem RGB e de profundidade e aplicado o detetor sobre elas. O corte da área de interesse é efetuado para diminuir a carga de informação a ser processada pelos detetores e diminuir a possibilidade de falsos positivos, causada por elementos do cenário. Com o uso deste algoritmo um dos requisitos é iniciar o sistema com o cenário sem pessoas ou objetos móveis.

4.2 Subtração de Fundo

Nas subsecções são apresentadas e descritas de forma mais pormenorizada as técnicas de subtração de fundo testadas e utilizadas. Na figura 4.2 ilustra a ideia de subtração de fundo.

4.2.1 *Mixture of Gaussian*

O *Mixture of Gaussian* (MOG) é um método de subtração de fundo com um sistema de atualização do modelo de fundo através de misturas gaussianas para determinar o valor do píxel do fundo [46, 47, 48]. Este algoritmo modela o valor de cada píxel por uma mistura de gaussianos. Com base na persistência e variância de cada gaussiano é determinado qual gaussiano deve corresponder ao valor de cada píxel do fundo. Os valores de píxeis que não entrarem nas distribuições gaussianas são considerados elementos não pertencentes ao fundo até que a sua persistência e variação caiam dentro da distribuição sendo então considerados parte do fundo. Desta forma este método consegue adaptar-se a alterações de iluminação e movimentos repetitivos de objetos e entrada e saída de objetos de cena [48].

Objetos com movimentos lentos demoram a ser integrados ao modelo de fundo porque a cor deles apresenta uma maior variância do que o fundo. Como o algoritmo depende de distribuições gaussianas, são necessárias várias imagens para o algoritmo começar a funcionar e obter valores para utilizar na construção do modelo de fundo.

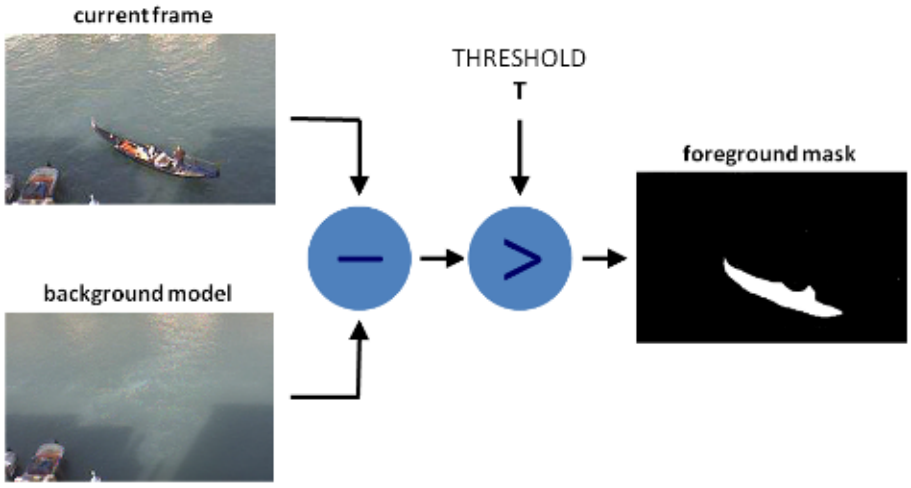


Figura 4.2: Conceito de subtração de fundo [49].

Este método apresenta bons resultados em algumas aplicações com alterações de iluminação e ambiente exterior e vários objetos em movimento ao mesmo tempo. Apesar desses bons resultados, devido à baixa velocidade de movimento e eventuais paragens das pessoas dentro de uma sala, este método apresenta alguns problemas em situações de interior com pessoas paradas.

O MOG aplicado e testado na proposta foi o implementado pelo OpenCV que segue o algoritmo apresentado em [46].

4.2.2 Algoritmo de Subtração de Fundo por diferença entre imagens

Devido a alguns problemas existentes no método de subtração de fundo MOG, nomeadamente a atualização do modelo de fundo que se um objeto estiver parado durante um certo tempo, como é habitual em situações de Ambiente Assistido, é considerado como parte do fundo, foi desenvolvido um algoritmo de subtração de fundo que considera um modelo de fundo estático e um algoritmo de subtração de fundo que combina informação de duas imagens e tem um modelo de fundo dinâmico. O algoritmo baseia-se no conceito de subtração de fundo, ou seja, uma subtração entre duas imagens sendo uma delas uma imagem definida pelo modelo de fundo. O algoritmo tem como base a criação de um modelo de fundo estático (que não é alterado ao longo do tempo) e a subtração do mesmo à imagem da qual se quer saber a existência de algum elemento diferente do fundo. Foi criado a pensar no uso em imagens de profundidade, mas foi testado nas imagens de profundidade e RGB.

Para a criação do modelo de fundo é admitido que no inicio do sistema, as primeiras imagens capturadas representam o cenário sem pessoas, ou seja, representam o que é considerado como fundo. Tanto a imagem do modelo de fundo como a imagem a testar são transformadas em imagens em escala de cinzento para que desta forma cada píxel seja representado por apenas um valor. De seguida é efetuada uma diferença entre as imagens. A diferença é calculada píxel a píxel, dando origem a uma nova imagem que representa o absoluto da diferença entre o modelo de

fundo e a imagem a testar. A imagem das diferenças passa de seguida por um *threshold* para eliminar valores de diferença abaixo do limite escolhido para a deteção de objetos diferentes do fundo, resultando numa imagem binária com "manchas" que representam possíveis movimentos de objetos ou pessoas. O valor de *threshold* utilizado foi determinado experimentalmente de forma a obter a melhor separação entre as pessoas e o fundo. Devido ao limite escolhido, ao ruído das imagens usadas e aos valores dos objetos a ser detetados serem muito próximos ao valor de fundo, as "manchas" não representam todo o corpo do objeto diferente do fundo. Por esse motivo a imagem das diferenças é filtrada de forma a aumentar a área das "manchas" e ligar aquelas que estão juntas, pois é provável representarem o mesmo objeto. No algoritmo 3 é apresentada a subtração por diferença entre imagens. Este algoritmo pode ser visto na forma de fluxograma nos anexos A.2 e na figura 4.9 um exemplo e imagens tipo que o algoritmo produz.

Algorithm 3 Subtração de Fundo por diferença entre imagens

```
    função BACKSUBTYPE1( imagem de entrada, limite)
2:      variavel diff
        variavel Imagem de saida
4:      se Modelo de Fundo não carregado então
            variavel modelo de fundo ← imagem de entrada
6:      senão
            enquanto todos os píxeis da imagem não forem percorridos faça
8:              diff ← abs(pxel img − pxel modelo de fundo)
                se diff < limite então
10:                 pixel imagem saida ← 0
                else
12:                 pixel imagem saida ← 1
                fim se
14:         fim enquanto
        fim se
16:     Filtragem da Imagem de saida
         devolve Imagem de saida
    fim função
```

4.2.3 Subtração de Fundo com Combinação de informação

Este algoritmo desenvolvido utilizando como base o Algoritmo de Subtração de Fundo apresentado anteriormente na secção 4.2.2, tem como principal diferença a fusão de informação da imagem de profundidade e RGB. O algoritmo foi criado para melhorar o comportamento e prestação do Algoritmo de Subtração de Fundo bem como a sua prestação e fiabilidade na deteção de diferenças no fundo.

O algoritmo recebe como entrada a imagem de profundidade e de RGB. De inicio aplica o primeiro algoritmo na imagem de profundidade resultando numa imagem de "manchas" que representam as diferenças entre o fundo e a imagem usada considerando apenas a profundidade. Esta imagem das "manchas" passa para a segunda parte do algoritmo, na qual vai ser efetuada a

subtração da imagem RGB. Como a subtração de fundo utilizando as imagens RGB produz elevado erro principalmente devido a diferenças de iluminação, é utilizada a imagem das "manchas" da primeira subtração (na imagem de profundidade) do algoritmo de forma a minimizar o erro nesta nova subtração e para fazer atualização do modelo de fundo do RGB. Como a subtração de fundo nas imagens de profundidade apresentam menos "manchas" do que nas imagens RGB, a subtração de fundo nas imagens RGB apenas vai ser efetuada onde existir "mancha" nas imagens de profundidade. Desta forma é eliminado algum erro devido a diferenças de iluminação existente na imagem RGB que não existe na imagem de profundidade. No algoritmo 4 é apresentada a subtração com combinação de informação. Este algoritmo pode ser visto na forma de fluxograma nos anexos A.3 e na figura 4.10 um exemplo do uso do algoritmo e imagens tipo que ele produz.

Algorithm 4 Subtração de Fundo com Combinação de informação

```
função BACKSUBTYPE2( imagem RGB, imagem Depth, limite RGB, limite Depth )
    variavel diff
    variavel Imagem de saida
    variavel img depth ← imagem Depth
    variavel img rgb ← imagem RGB
    sub depth ← BackSubType1(img depth, limite Depth)
    se Modelo de Fundo RGB não carregado então
        modelo de fundo RGB ← img rgb
    senão
        enquanto todos os pixels da imagem não forem percorridos faça
            diff ← abs(pixel img rgb − pixel modelo de fundo RGB)
            se pixel sub depth == 1 então
                se diff < limite então
                    pixel imagem saida ← 0
                senão
                    pixel imagem saida ← 1
                fim se
            senão
                pixel modelo de fundo RGB ← pixel img rgb
            fim se
        fim enquanto
    fim se
    Segmentação por Crescimento de Regiões
    Filtragem da Imagem de saida
    devolve Imagem de saida
fim função
```

Sendo a mudança da iluminação o principal problema da subtração de fundo com imagens RGB, para além da subtração ser assistida pela subtração da imagem de profundidade, o modelo de fundo usado nas imagens RGB é atualizado. A atualização do modelo de fundo RGB é efetuada nos píxeis onde a subtração de fundo da imagem de profundidade diz não existir movimento, desta forma, o modelo de fundo RGB é atualizado para o novo valor RGB nas áreas onde não existe movimento de pessoas.

A figura 4.3 demonstra a aplicação da Subtração de Fundo por diferença entre imagens na imagem RGB e de profundidade e a Subtração de Fundo com combinação de informação na mesma *frame*. Nesta figura é possível identificar diferenças entre as subtrações e concluir que a combinação de informação é capaz de obter melhores resultados.

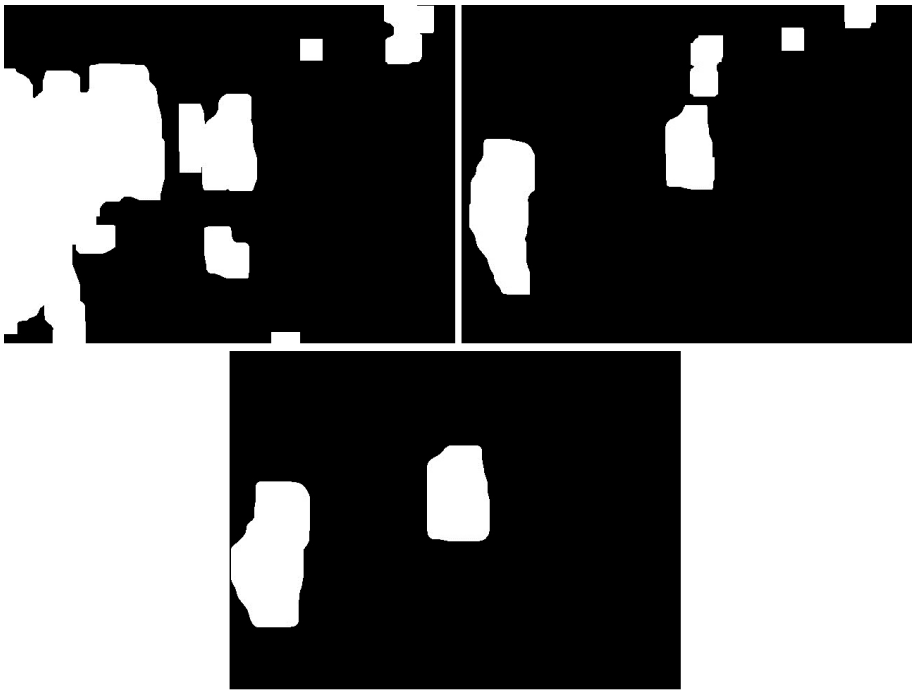


Figura 4.3: Exemplo de subtração de fundo pelos dois algoritmos. A imagem do canto superior esquerdo ilustra o resultado do algoritmo de subtração com imagem RGB e a imagem do canto superior direito o resultado com imagem de profundidade. A imagem presente a meio representa o resultado da junção de ambas subtrações.

Uma vez executada a subtração da imagem RGB assistida pela imagem de profundidade, é efetuada uma nova fusão de informação, na qual a imagem resultante da subtração do RGB ajuda na filtragem da imagem da subtração de fundo da imagem de profundidade. Esta fusão é efetuada para eliminar algum erro existente na imagem de profundidade, dando a imagem final do algoritmo que passa pela mesma filtragem que é usada no primeiro algoritmo. A fusão é efetuada utilizando uma lógica de segmentação por crescimento de regiões, na qual a imagem da subtração da imagem RGB representa as sementes a serem usadas, e a imagem da subtração da imagem de profundidade representa a imagem a ser segmentada. Desta forma a imagem resultante apenas vai incluir as "manchas" que existirem em ambas imagens da subtração, ou seja, apenas as "manchas" da imagem de profundidade que tiverem pelo menos um ponto das "manchas" da imagem de RGB são incluídas na imagem final, todas as "manchas" da imagem de profundidade que não incluir algum ponto são eliminadas. A figura 4.4 ilustra a ideia de fusão de informação utilizando uma lógica de segmentação por crescimento de regiões para eliminar as "manchas" na imagem de profundidade.

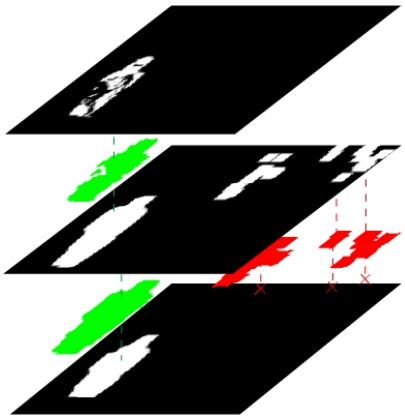


Figura 4.4: Exemplo ilustrativo do Crescimento de Regiões. De cima para baixo, a primeira imagem representa o resultado da subtração de fundo da imagem RGB assistida pela imagem de profundidade, que serve de sementes para determinar as regiões da imagem do meio que devem estar presentes na imagem final, representada pela ultima imagem.

4.3 Detetores

Na presente secção são apresentados os detetores estudados e testados para efetuar a deteção de pessoas. Inicialmente é apresentado o detetor HOG e o detetor de "Ombro-Cabeça-Ombro" seguido pelo detetor de Cara e por fim o detetor de pele.

4.3.1 *Histogram of Oriented Gradients*

O detetor HOG [29] como referido na secção 2.2.5 é um descritor de objetos utilizados para detetar vários objetos e detetar pessoas.

O descritor começa por filtrar a imagem de forma a melhorar o desempenho. Uma vez filtrada a imagem, é dividida em pequenas células e calculado o histograma dos gradientes de cada píxel da célula. Para o cálculo dos gradientes são usadas matrizes coluna ou linha (consoante a direção que se quer determinar o gradiente) para calcular a primeira derivada entre os píxeis. O histograma pode ter os valores distribuídos de 0^o a 180^o chamado *"unsigned"* ou de 0^o a 360^o sendo chamado *"signed"*, e dividido em vários canais como já referido em 2.2.5. Os histogramas criados são normalizados de forma a melhorar o descritor. O ajuste é efetuado através de um valor de intensidade calculado usando uma área maior do que uma célula, chamado bloco. O bloco é constituído por várias células. Para melhorar a normalização, os blocos podem ser sobrepostos de forma à mesma célula contribuir em blocos diferentes. Os parâmetros de sobreposição, tamanho de células e blocos são escolhidos conforme o objeto a descrever.

O detetor implementado foi o existente na biblioteca de funções do OpenCV, utilizando células 8×8; o tamanho dos blocos é 16×16 e 9 canais no histograma "unsigned". O HOG só por si não classifica o objeto, apenas cria o descritor. O vetor de valores que descreve o objeto tem de ser classificado. O classificador utilizado é SVM. O SVM presente no OpenCV já inclui um conjunto

de dados de treino, ou seja, está pronto a utilizar sem ser necessário treinar o sistema, mas existe a possibilidade de treinar o sistema. Na figura 4.5 esta representado o resultado da aplicação do detetor HOG utilizado na proposta nos primeiros testes ao detetor.

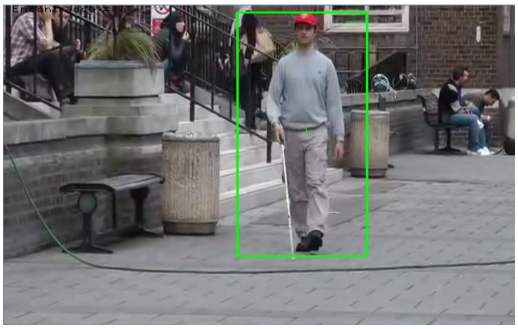


Figura 4.5: Primeiro teste do HOG do OpenCV em que a caixa verde representa o que o detetor considerou como pessoa.

4.3.2 "Ombro-Cabeça-Ombro"

Este detetor foi considerado e utilizado por efetuar a deteção de pessoas pois como utiliza a parte superior do corpo para efetuar a deteção, apresenta bons resultados em situações de ocultação que geralmente acontecem na parte inferior do corpo.

O detetor de *Upper Body*, ou seja, detetor de "Ombro-Cabeça-Ombro", utiliza *Haar feature* para efetuar a descrição da imagem. O descritor *Haar feature* é calculado utilizando um *kernel* constituído por uma zona representada por zonas a preto e outras a branco como apresentado na figura 4.6. O valor do descritor é calculado passando um dos *kernels* por cima da imagem e calculando a subtração da soma de todos os píxeis da imagem cobertos pela área branca pela soma dos píxeis da zona coberta pela área a preto [50].

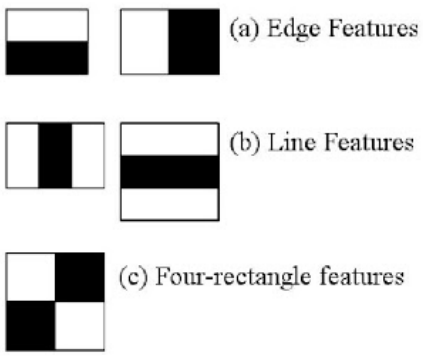


Figura 4.6: *Haar-feature*, retirado de [50].

Após efetuado o descritor é necessário efetuar a classificação da informação. A classificação é efetuada utilizando um *Cascade of Classifiers*. O *Cascade of Classifiers* classifica a informação passando a mesma por várias classificações ao longo de uma "cascata" de etapas de classificação. Em cada etapa da "cascata" a informação é classificada e é imediatamente descartada por não apresentar neste caso um conjunto "ombro-cabeça-ombro" ou passa para a etapa seguinte onde aplica outra classificação, sucessivamente até ao fim onde classifica como "ombro-cabeça-ombro" neste caso. O classificar presente no OpenCV inclui uma "cascata" já treinada que é usada pelo algoritmo.

4.3.3 Cara

O detetor de Cara foi utilizado por apresentar bons resultados na deteção de caras perto da câmara e desta forma ajudar a eliminar algum falso positivo do detetor HOG ou "Ombro-Cabeça-Ombro". Este detetor utiliza a mesma lógica que o detetor de "Ombro-Cabeça-Ombro". O *Haar-feature* é invariante à aplicação podendo ser usado para descrever diferentes tipos de objetos. A diferença entre este detetor e o detetor de "Ombro-Cabeça-Ombro" reside apenas nos valores de treino do classificado *Cascade of Classifiers*. O OpenCV por sua vez também já possui um classificar treinado com a finalidade de detetar caras.

Na figura 4.7 esta representado o resultado da aplicação do detetor "Ombro-Cabeça-Ombro" (imagem da esquerda) e de Cara (imagem da direita) utilizado na proposta nos primeiros testes aos detetores.

Figura 4.7: Exemplo de teste do detetor de *Upper Body* e Cara. Na imagem da esquerda podemos ver representado com um retângulo vermelho a área detetada como *Upper Body*, e na imagem da direita representado com um retângulo azul a área detetada como cara.

4.3.4 Pele

Foi implementado e testado um detetor de pele baseado em HSV. A utilização deste detetor tinha como fim tentar diminuir os falsos positivos causados pelos outros detetores. Este detetor apresentado em [51] tem como principio a conversão da imagem em espaço de cores RGB para uma imagem com espaço de cores em HSV. Uma vez a imagem em HSV, é utilizado apenas o

canal H que representa as cores e é colocado a "1" todos os píxeis que se encontram dentro dos limites de cor da pele e a "0" os restantes. Os limites apresentados e usados foram entre 6 e 38.

Devido ao ruído presente no resultado deste detetor, causado pela cor do cenário e devido ao facto das pessoas não apresentarem sempre pele visível, ele não foi utilizado. Na figura 4.8 está apresentado a aplicação do detetor de Pele testado à imagem da esquerda, sendo o resultado a imagem da direita.

Figura 4.8: Teste do detetor de pele.

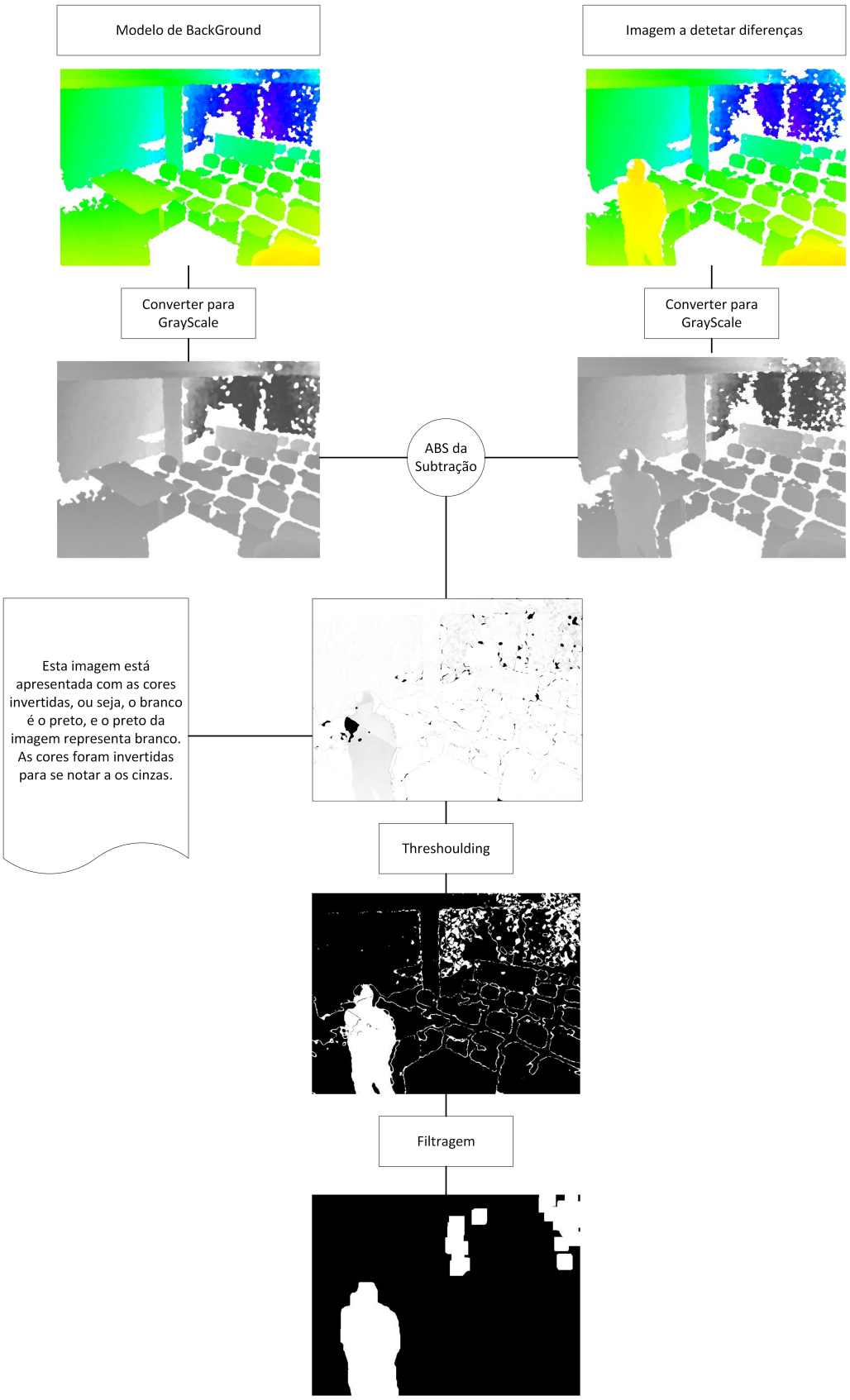


Figura 4.9: Exemplo de funcionamento do Algoritmo de Subtração de Fundo por diferença entre imagens (4.2.2).

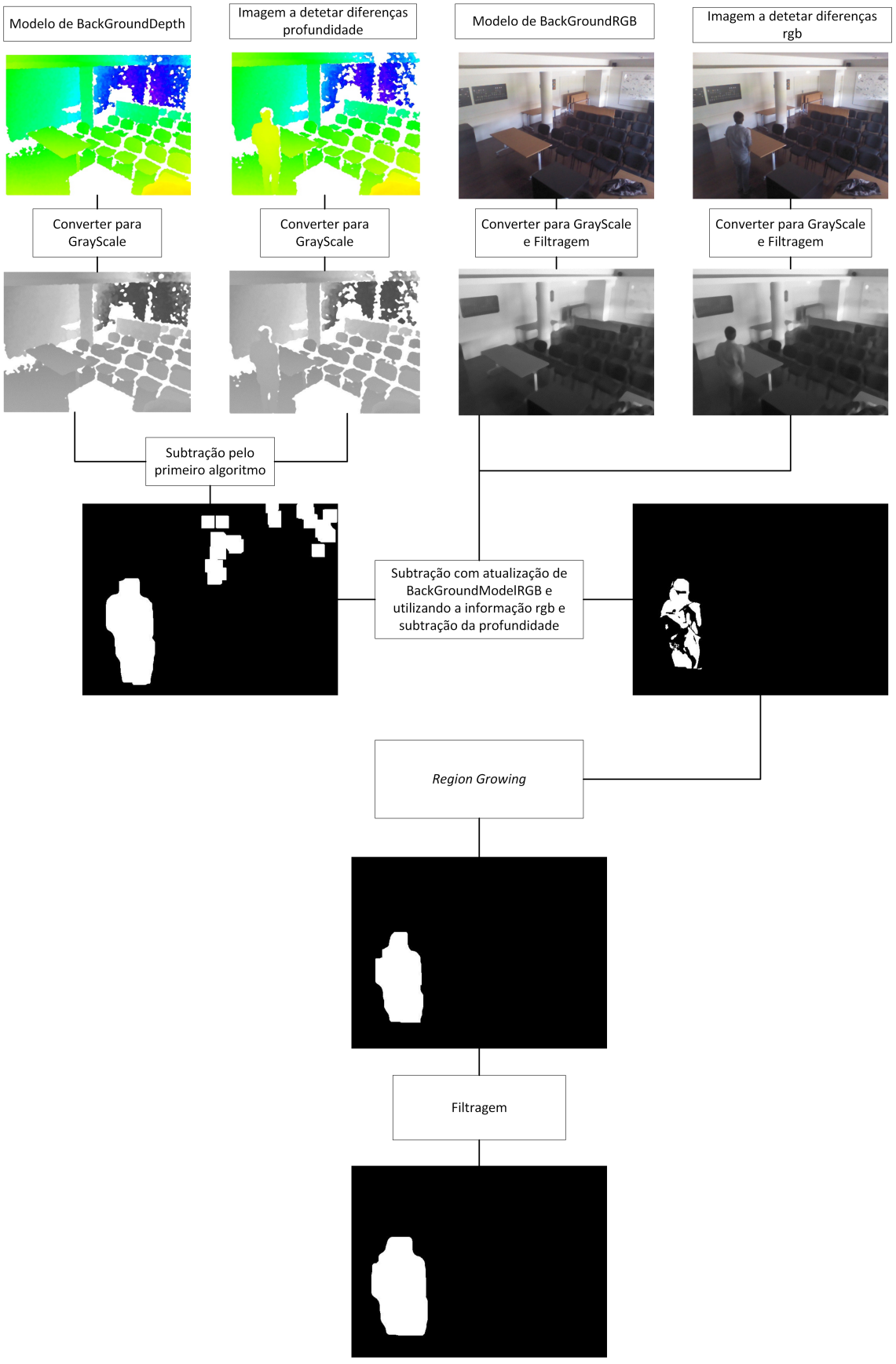


Figura 4.10: Exemplo de funcionamento do Algoritmo de Subtração de Fundo com Combinação de informação (4.2.3).

Capítulo 5

Resultados

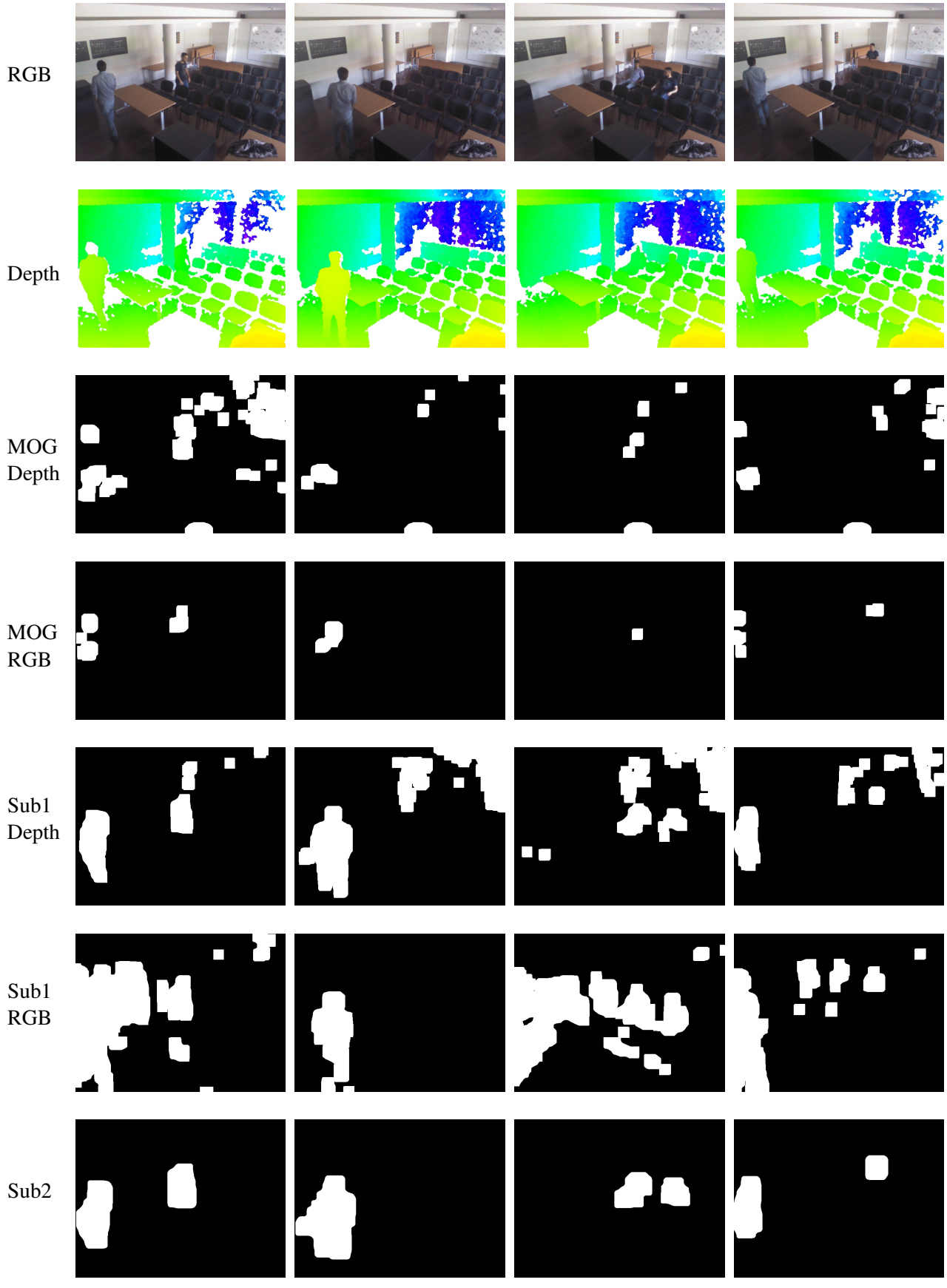
Neste capítulo são apresentados os resultados da proposta aplicada a 3 conjuntos de capturas. Na primeira secção são demonstrados os resultados dos sistemas de subtração testados e criados de forma a comparar os resultados entre eles. Na última secção são apresentados os cenários utilizados e os resultados referentes à aplicação do algoritmo proposto e dos detetores sem o algoritmo proposto.

Para cada um dos cenários são apresentados dados referentes às deteções. Os dados apresentados em primeiro lugar referem-se aos resultados da aplicação dos detetores HOG, "Ombro-Cabeça-Ombro" (*Upper Body*) e Cara às imagens RGB sem a utilização do algoritmo proposto. Os dados apresentados em segundo lugar referem-se aos resultados da aplicação do algoritmo proposto de subtração de fundo ao conjunto de capturas. Por fim, é apresentada a fusão dos resultados dos detetores com o subtrator de fundo, pelo algoritmo proposto.

5.1 Subtração de Fundo

Na tabela seguinte 5.1 está apresentado o resultado da aplicação das várias técnicas de subtração de fundo estudadas e implementadas. Este conjunto de imagens foi criado e exposto para comparação dos resultados entre as várias técnicas de subtração. As primeiras duas linhas da tabela são as imagens de RGB e profundidade. A terceira e quarta linha representam o resultado do método MOG aplicado às imagens de profundidade e RGB apresentadas nas primeiras linhas da tabela. A quinta e sexta linha representam o resultado da subtração de fundo sem fusão de informação com o nome de "Sub1 Depth" e "Sub1 RGB". A última linha da tabela apresenta o resultado da subtração de fundo implementada no algoritmo proposto sugerido, com o nome "Sub2".

Tabela 5.1: Testes comparativos de Subtração de Fundo.



5.2 Conjuntos de capturas

Nesta secção são apresentados três cenários que foram utilizados para efetuar os testes dos detetores e do algoritmo proposto. Os primeiros dois cenários apresentam apenas uma pessoa na sala e o primeiro é o que apresenta menores dimensões. O terceiro cenário apresenta uma sala de maiores dimensões em relação às outras duas e nele estão presentes duas pessoas.

5.2.1 Cenário 1

Estas capturas apresentam uma pessoa numa sala de pequenas dimensões, figura 5.1. A pessoa percorre a sala, apresentando situações sem ocultações e situações com alguma ocultação, devido a objetos presentes em cena.

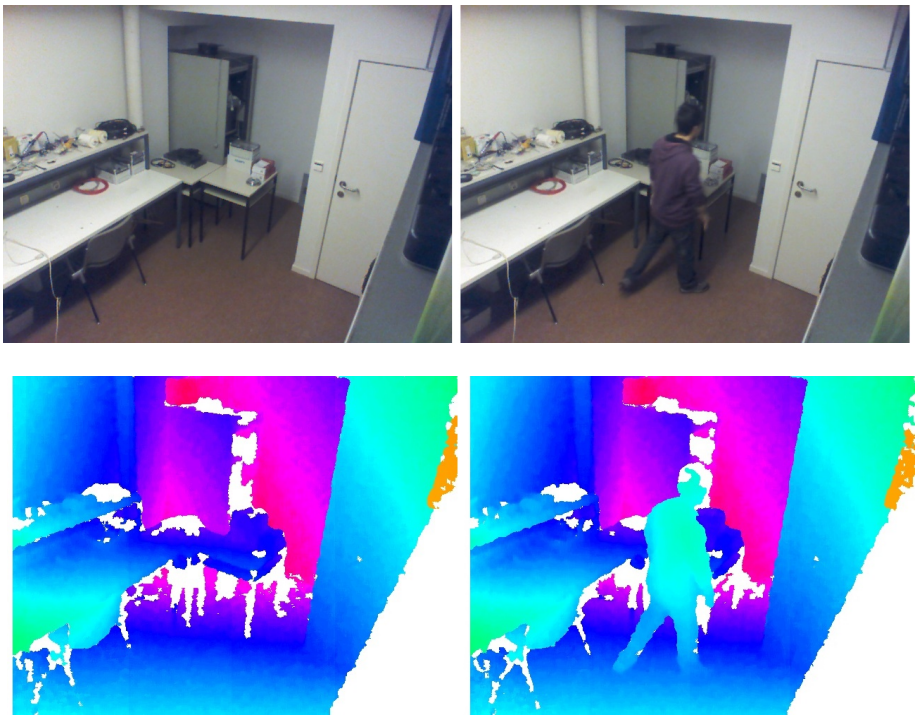


Figura 5.1: Cenário do Conjunto de capturas 1. Na coluna da esquerda está ilustrado a imagem RGB e de profundidade do cenário sem pessoas, ou seja, o que é considerado como fundo. Na coluna da direita está apresentado uma pessoa a percorrer o cenário.

Na tabela 5.2 é apresentado os resultados das métricas *False Alarme Rate* (FAR), *Detection Rate* (DR) e *Positive Prediction* (PP) para os detetores HOG, "Ombro-Cabeça-Ombro" (*Upper Body*) e de Cara sem a utilização de subtração de fundo e na figura 5.2 apresenta as deteções em algumas *frames*. A negrito é apresentado os melhores resultados para cada métrica.

Tabela 5.2: Resultados dos detetores na imagem RGB da Cenário 1 sem Subtração de Fundo.

	HOG	*Upper Body*	Cara
FAR	74.3%	65.6%	**10.2%**
DR	34.0%	**72.9%**	44.7%
PP	25.7%	34.4%	**89.8%**

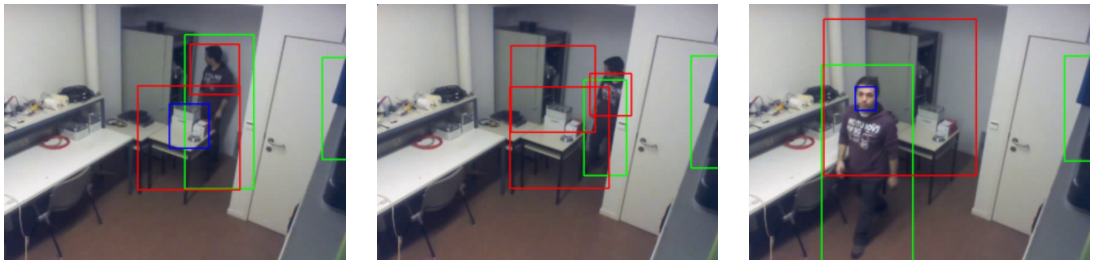


Figura 5.2: Resultados dos detetores na imagem RGB do Cenário 1 sem Subtração de Fundo. Cada caixa representa a deteção por parte de um dos detetores. A vermelho está representada a deteção por parte do detetor de "Ombro-Cabeça-Ombro", a azul o detetor de Cara e a verde o detetor HOG.

Na tabela 5.3 e figura 5.3 são apresentado os resultados referentes à aplicação do algoritmo proposto sem a fusão de informação. A negrito é apresentado os melhores resultados para cada métrica. Na coluna "Fundo" é apresentado os resultados das métricas considerando a subtração de fundo como um detetor de pessoas. Estes valores apresentam a capacidade e precisão com que a subtração de fundo detetou o movimento de pessoas no cenário. A coluna HOG e *Upper Body* apresentam os valores dos detetores nas imagens RGB e de profundidade individualmente. A última coluna apresenta os dados da aplicação do detetor de Cara na imagem RGB, visto que na imagem de profundidade não existem deteções positivas por não ser possível visualizar a cara.

Figura 5.3: Resultados dos detetores na imagem RGB e profundidade do Cenário 1 com Subtração de Fundo. Cada caixa representa a deteção por parte de um dos detetores. A vermelho está representada a deteção por parte do detetor de "Ombro-Cabeça-Ombro", a azul o detetor de Cara e a verde o detetor HOG.

Tabela 5.3: Resultados dos detetores na imagem RGB e profundidade da Cenário 1 com Subtração de Fundo.

	Fundo	HOG		*Upper Body*		Cara
		RGB	*Depth*	RGB	*Depth*	RGB
FAR	7.0%	**0.0%**	16.0%	32.4%	28.2%	6.7%
DR	99.8%	4.1%	13.2%	**59.2%**	9.5%	43.9%
PP	92.9%	**100%**	83.9%	67.6%	71.8%	93.3%

Na tabela 5.4 na coluna "Todos detetores" é apresentada o resultado da junção de todos os detetores na imagem RGB como se fossem apenas um e a coluna "Combinação de detetores" apresenta a combinação das deteções na imagem RGB e profundidade com eliminação de deteções sobrepostas. A última coluna apresenta o detetor com melhor taxa de deteção da tabela 5.3. A negrito é apresentado os melhores resultados para cada métrica. A figura 5.4 apresenta os resultados da combinação de detetores em algumas *frames*.

Tabela 5.4: Resultados da fusão de informação da Cenário 1.

	Algoritmo		*Upper Body*
	Todos detetores	Combinação de detetores	RGB
FAR	50.1%	43.0%	**32.4%**
DR	68.3%	**71.3%**	59.2%
PP	49.3%	57.0%	**67.6%**

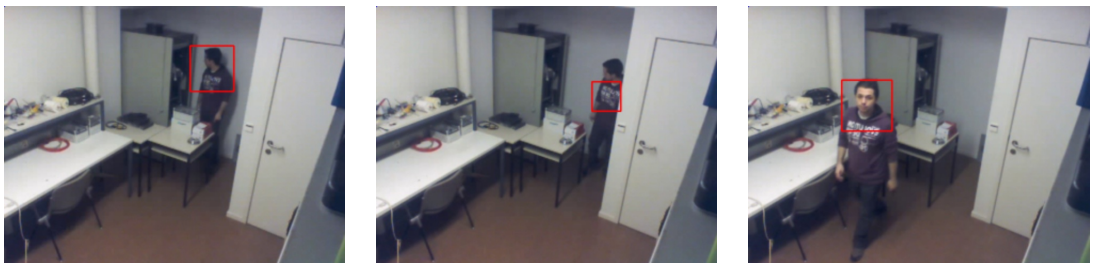
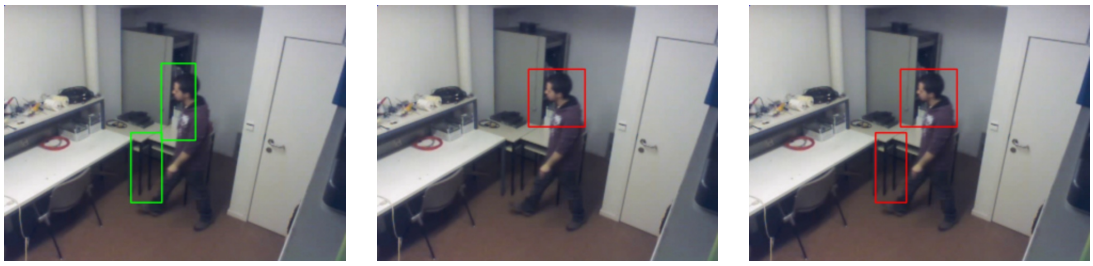


Figura 5.4: Resultados da fusão dos detetores no cenário 1 com Subtração de Fundo. Cada caixa representa a deteção por parte da fusão de informação.

Com a análise da tabela 5.4, conclui-se que com a fusão de informação do algoritmo proposto conseguem-se mais deteções positivas em relação à aplicação dos detetores em separado na imagem RGB, com subtração de fundo. Neste caso, a precisão da deteção (PP) é mais baixa devido ao aumento de falsos positivos, causados pelos detetores nas imagens de profundidade, que devido ao erro apresentam deteções erradas como ilustrado na figura 5.5. A imagem apresenta: a) o resultado dos detetores na imagem de profundidade; b) resultado dos detetores na imagem RGB; c)

resultado da fusão das informações, em que mostra um falso positivo introduzido pela informação da imagem de profundidade.



a) b) c)

Figura 5.5: Resultados da fusão dos detetores no cenário 1, mostra de erro. Cada caixa representa a deteção por parte da fusão de informação.

5.2.2 Cenário 2

Estas capturas apresentam uma pessoa numa sala, figura 5.6. A pessoa percorre a sala, apresentando situações sem ocultações e situações de elevada ocultação devido a objetos presentes. Este cenário apresenta uma área com muito ruído na imagem de profundidade. O ruído presente permitiu observar o comportamento dos detetores e do algoritmo proposto em situações onde as imagens de profundidade apresentam grandes áreas de ruído, devido ao cenário em que o sistema está a ser utilizado.

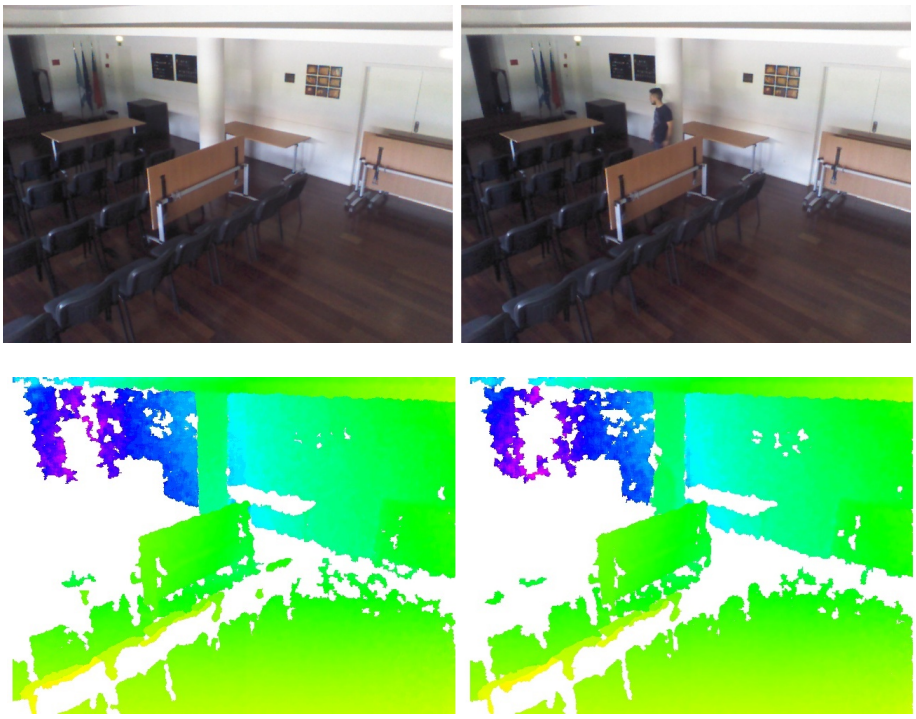


Figura 5.6: Cenário do Conjunto de capturas 2. Na coluna da esquerda está ilustrado a imagem RGB e de profundidade do cenário sem pessoas, ou seja, o que é considerado como fundo. Na coluna da direita está apresentado uma pessoa a percorrer o cenário.

Na tabela 5.5 é apresentado os resultados das métricas *False Alarme Rate* (FAR), *Detection Rate* (DR) e *Positive Prediction* (PP) para os detetores HOG, "Ombro-Cabeça-Ombro" (*Upper Body*) e de Cara sem a utilização de subtração de fundo e na figura 5.7 apresenta as deteções em algumas *frames*. A negrito é apresentado os melhores resultados para cada métrica.

Tabela 5.5: Resultados dos detetores na imagem RGB da Cenário 2 sem Subtração de Fundo.

	HOG	*Upper Body*	Cara
FAR	84.3%	87.7%	**9.4%**
DR	41.4%	**58.4%**	12.8%
PP	15.6%	12.3%	**90.6%**

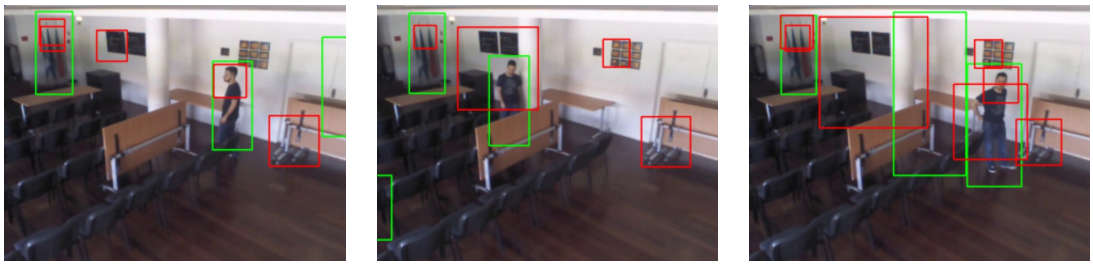


Figura 5.7: Resultados dos detetores nas imagens RGB do Cenário 2 sem Subtração de Fundo. Cada caixa representa a deteção por parte de um dos detetores. A vermelho está representada a deteção por parte do detetor de "Ombro-Cabeça-Ombro", a azul o detetor de Cara e a verde o detetor HOG.

Na tabela 5.6 e figura 5.8 são apresentado os resultados referentes à aplicação do algoritmo proposto sem a fusão de informação. A negrito é apresentado os melhores resultados para cada métrica. Na coluna "Fundo" é apresentado os resultados das métricas considerando a subtração de fundo como um detetor de pessoas. Estes valores apresentam a capacidade e precisão com que a subtração de fundo detetou o movimento de pessoas no cenário. A coluna HOG e *Upper Body* apresentam os valores dos detetores nas imagens RGB e de profundidade individualmente. A última coluna apresenta os dados da aplicação do detetor de Cara na imagem RGB, visto que na imagem de profundidade não existem deteções positivas por não ser possível visualizar a cara.

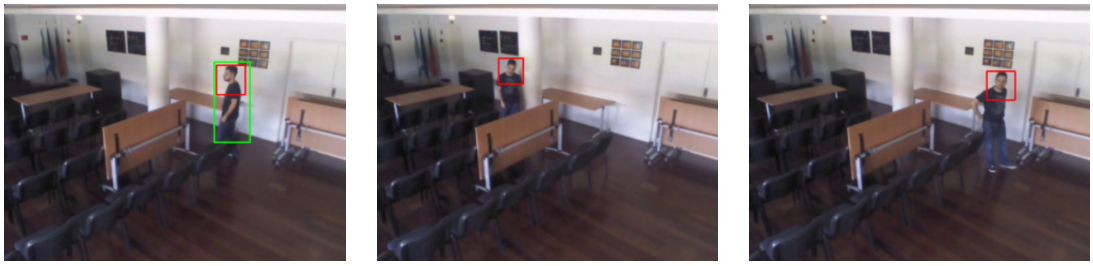


Figura 5.8: Resultados dos detetores nas imagens RGB e profundidade do Cenário 2 com Subtração de Fundo. Cada caixa representa a deteção por parte de um dos detetores. A vermelho está representada a deteção por parte do detetor de "Ombro-Cabeça-Ombro", a azul o detetor de Cara e a verde o detetor HOG.

Tabela 5.6: Resultados dos detetores na imagem RGB e profundidade do Cenário 2 com Subtração de Fundo.

	Fundo	HOG		*Upper Body*		Cara
		RGB	*Depth*	RGB	*Depth*	RGB
FAR	16.5%	**0.0%**	12.8%	19.1%	8.7%	4.7%
DR	92.9%	4.1%	4.2%	**64.0%**	11.2%	12.6%
PP	83.5%	**100%**	87.2%	80.8%	91.3%	95.3%

Na tabela 5.7 na coluna "Todos detetores" é apresentada o resultado da junção de todos os detetores na imagem RGB como se fossem apenas um e a coluna "Combinação de detetores" apresenta a combinação das deteções na imagem RGB e profundidade com eliminação de deteções sobrepostas. A última coluna apresenta o detetor com melhor taxa de deteção da tabela 5.6. A negrito é apresentado os melhores resultados para cada métrica. A figura 5.9 apresenta os resultados da combinação de informação em algumas *frames*.

Tabela 5.7: Resultados da fusão de informação da Cenário 2.

	Algoritmo		*Upper Body*
	Todos detetores	Combinação de detetores	RGB
FAR	31.6%	19.9%	**19.1%**
DR	66.0%	**66.4%**	64.0%
PP	68.4%	80.1%	**80.8%**

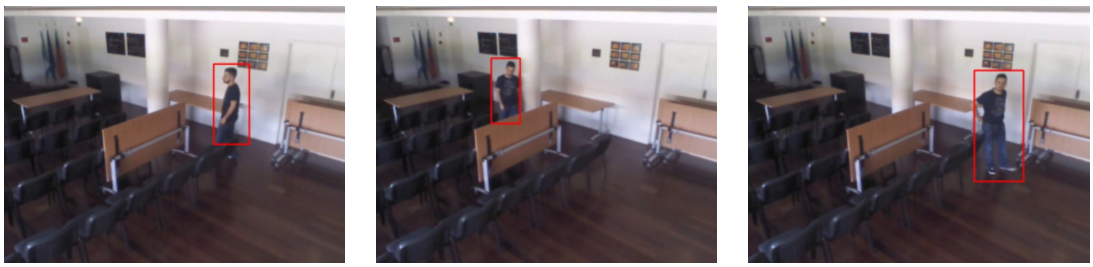
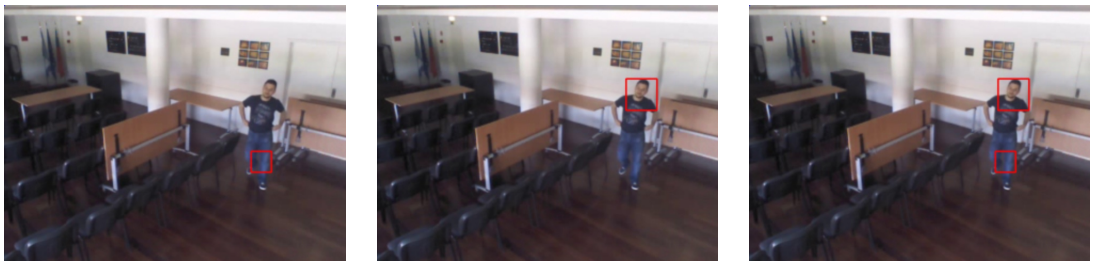


Figura 5.9: Resultados da fusão dos detetores no cenário 2 com Subtração de Fundo. Cada caixa representa a deteção por parte da fusão de informação.

Com a análise da tabela 5.7 conclui-se que com a fusão de informação do algoritmo proposto consegue maior taxa de deteção em relação à aplicação dos detetores em separado na imagem RGB com subtração de fundo, como aconteceu com o cenário 1. Neste caso, tal como aconteceu no cenário 1, a precisão da deteção (PP) é mais baixa devido ao aumento de falsos positivos causados pelos detetores nas imagens de profundidade devido ao erro da imagem, como é ilustrado na figura 5.10. A imagem apresenta: a) o resultado dos detetores na imagem de profundidade; b) resultado dos detetores na imagem RGB; c) resultado da fusão das informações, em que mostra um falso positivo introduzido pela informação da imagem de profundidade.



a) b) c)

Figura 5.10: Resultados da fusão dos detetores no cenário 2, mostra de erro. Cada caixa representa a deteção por parte da fusão de informação.

5.2.3 Cenário 3

Estas capturas apresentam duas pessoas numa sala, figura 5.11. Uma das pessoas anda do meio da sala para o ponto mais afastado da câmara, apresentando quase sempre algum nível de ocultação. A segunda pessoa anda entre a localização da câmara e o meio da sala apresentando um menor número de ocultações e de cruzamento com a segunda pessoa no meio da sala.

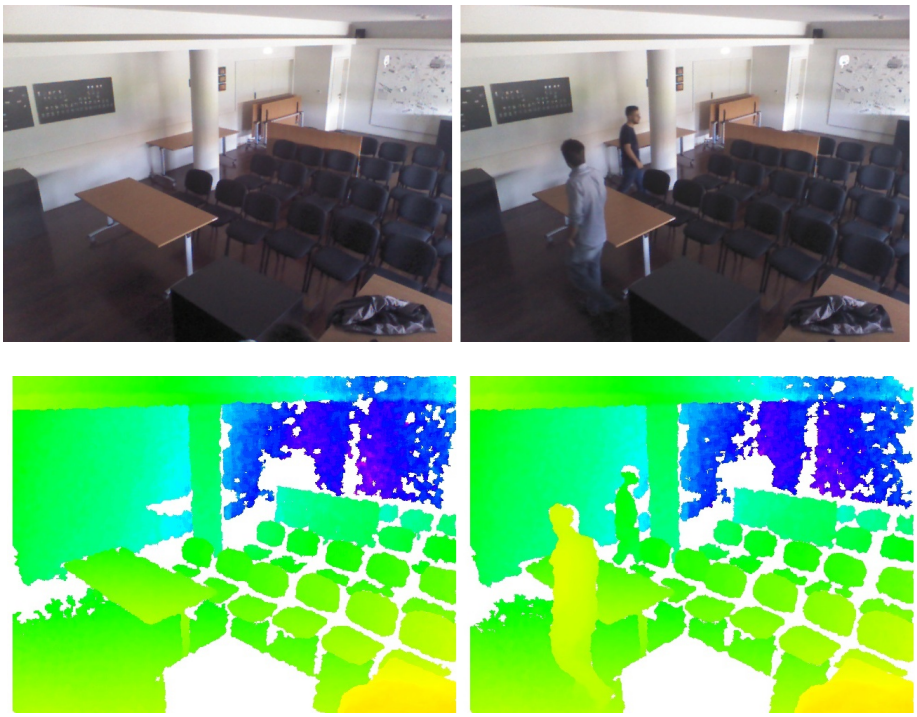


Figura 5.11: Cenário do Conjunto de capturas 3. Na coluna da esquerda está ilustrado a imagem RGB e de profundidade do cenário sem pessoas, ou seja, o que é considerado como fundo. Na coluna da direita está apresentado duas pessoas a percorrer o cenário.

Na tabela 5.8 é apresentado os resultados das métricas *False Alarme Rate* (FAR), *Detection Rate* (DR) e *Positive Prediction* (PP) para os detetores HOG, "Ombro-Cabeça-Ombro" (*Upper Body*) e de Cara sem a utilização de subtração de fundo e na figura 5.12 apresenta as deteções em algumas *frames*. A negrito é apresentado os melhores resultados para cada métrica.

Tabela 5.8: Resultados dos detetores na imagem RGB da Cenário 3 sem Subtração de Fundo.

	HOG	*Upper Body*	Cara
FAR	**30.2%**	57.6%	39.7%
DR	32.9%	**70.3%**	43.1%
PP	25.7%	34.4%	**89.8%**

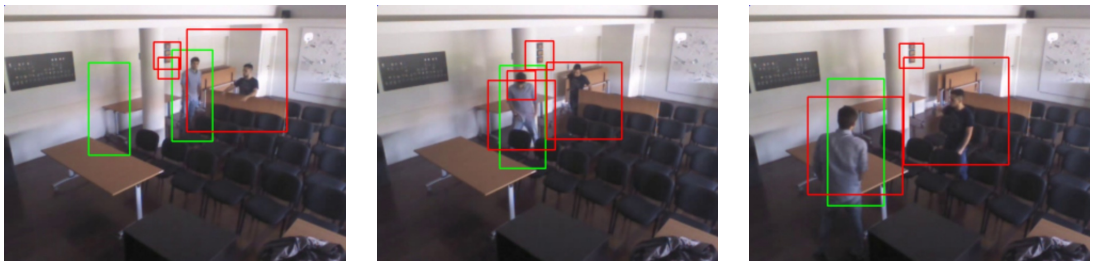
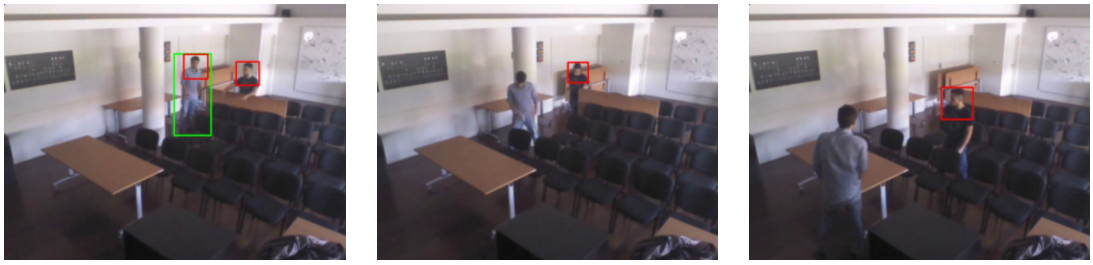


Figura 5.12: Resultados dos detetores nas imagens RGB do Cenário 3 sem Subtração de Fundo. Cada caixa representa a deteção por parte de um dos detetores. A vermelho está representada a deteção por parte do detetor de "Ombro-Cabeça-Ombro", a azul o detetor de Cara e a verde o detetor HOG.

Na tabela 5.9 e figura 5.13 são apresentado os resultados referentes à aplicação do algoritmo proposto sem a fusão de informação. A negrito é apresentado os melhores resultados para cada métrica. Na coluna "Fundo" é apresentado os resultados das métricas considerando a subtração de fundo como um detetor de pessoas. Estes valores apresentam a capacidade e precisão com que a subtração de fundo detetou o movimento de pessoas no cenário. A coluna HOG e *Upper Body* apresentam os valores dos detetores nas imagens RGB e de profundidade individualmente. A última coluna apresenta os dados da aplicação do detetor de Cara na imagem RGB, visto que na imagem de profundidade não existem deteções positivas por não ser possível visualizar a cara.



a) b) c)

Figura 5.13: Resultados dos detetores nas imagens RGB do Cenário 3 com Subtração de Fundo. Cada caixa representa a deteção por parte de um dos detetores. A vermelho está representada a deteção por parte do detetor de "Ombro-Cabeça-Ombro", a azul o detetor de Cara e a verde o detetor HOG.

Na tabela 5.10 na coluna "Todos detetores" é apresentada o resultado da junção de todos os detetores na imagem RGB como se fossem apenas um e a coluna "Combinação de detetores" apresenta a combinação das deteções na imagem RGB e profundidade com eliminação de deteções sobrepostas. A última coluna apresenta o detetor com melhor taxa de deteção da tabela 5.9. A negrito é apresentado os melhores resultados para cada métrica. Com a análise da tabela 5.10 conclui-se que no cenário 3, a fusão de informação pelo algoritmo proposto apresenta melhores

Tabela 5.9: Resultados dos detetores na imagem RGB e profundidade da Cenário 3 com Subtração de Fundo.

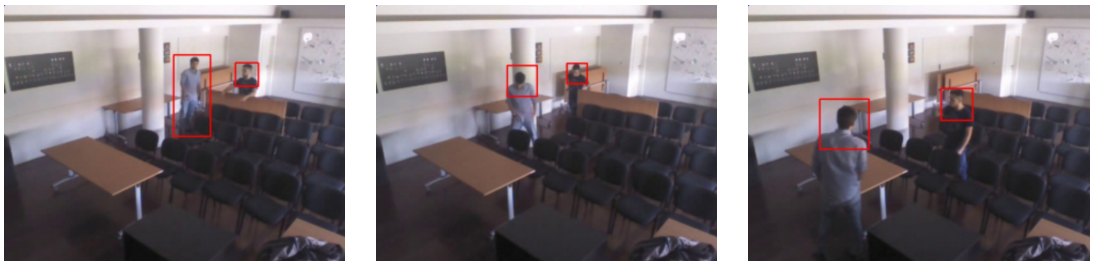
	Fundo	HOG		*Upper Body*		Cara
		RGB	*Depth*	RGB	*Depth*	RGB
FAR	19.0%	13.0%	**12.2%**	13.5%	17.3%	12.5%
DR	96.2%	4.0%	1.1%	**64.0%**	4.0%	0.6%
PP	81.0%	87.0%	**87.7%**	86.5%	82.6%	87.5%

resultados nos três aspetos, comparativamente ao detetor individual, que apresenta melhor taxa de deteção.

Tabela 5.10: Resultados da fusão de informação do Cenário 3.

	Algoritmo		*Upper Body*
	Todos detetores	Combinação de detetores	RGB
FAR	17.0%	**12.5%**	13.5%
DR	66.0%	**66.8%**	64.0%
PP	83.0%	**87.5%**	86.5%

A figura 5.14 apresenta três imagens do cenário 3 com as deteções por parte da fusão de informação do algoritmo proposto. Comparando a figura 5.14 com a figura 5.13 conclui-se que a fusão de informação entre deteções em RGB e profundidade produz maior número de deteções positivas. Apesar do número de deteções nas imagens de profundidade ser baixo, a junção desta informação com a informação das imagens RGB produz um aumento da taxa de deteção, conseguindo assim obter um maior número de deteções de pessoas que não tenham sido apresentadas nas imagens RGB. A figura 5.14 b) apresenta a mesma *frame* que a 5.13 b) onde se constata que a fusão de informação deteta as duas pessoas, enquanto que nas imagens RGB apenas foi detetada uma pessoa.



a) b) c)

Figura 5.14: Resultados da fusão dos detetores no cenário 3 com Subtração de Fundo. Cada caixa representa a deteção por parte da fusão de informação.

Capítulo 6

Conclusões

Neste Capítulo é feita uma discussão final da dissertação realizada e algumas sugestões para um trabalho futuro de forma a melhorar e ampliar o algoritmo proposto.

6.1 Conclusões Finais

Nesta dissertação foi realizado um estudo e implementação de um algoritmo de deteção de pessoas com algum nível de ocultação utilizando o *Kinect* num cenário de Ambiente Assistido. Este estudo compreendeu dois momentos distintos relativamente ao seu desenvolvimento. No primeiro, foi elaborado um estudo sobre a informação e caraterísticas do *Kinect for Windows*; no segundo momento realizou-se o estudo e implementação de um algoritmo para detetar pessoas com algum nível de ocultação. Com o estudo do *Kinect* pude concluir que as imagens produzidas apresentam-se desalinhadas e que a imagem de profundidade apresenta algum ruído devido à influência da luz. A malha de infra-vermelho projetada pelo *Kinect* apresenta situações de falha devido à cor na gama do preto de alguns objetos, reflexão da luz em algumas superfícies e efeito de sombra. A distância máxima captada durante a criação das sequências pelo *Kinect* foi de 12m. Apresenta imunidade ao vidro e boa robustez à variação de iluminação.

Na implementação do algoritmo proposto e na realização de testes concluí que o algoritmo apresenta melhor resultado na taxa de deteção em comparação a testes dos detetores sem o algoritmo. A melhoria deve-se à diminuição da área de processamento e à filtragem de áreas sem potenciais pessoas. Os resultados dos detetores HOG e detetor de face aplicados à imagem RGB, como era esperado, apresentam baixa taxa de deteção em relação ao detetor *Upper Body* que é mais talhado para detetar pessoas em situação de ocultação.

A partir da análise dos dados do algoritmo concluí que o detetor HOG apresenta baixa taxa de deteção em situação de ocultações de alguma parte do corpo, como era esperado, devido ao facto do HOG estar treinado para detetar pessoas com informação de corpo inteiro. Com a aplicação do algoritmo proposto, a taxa de deteções aumenta em relação à aplicação dos detetores sem a subtração de fundo à imagem RGB. No caso do cenário 1 e 2 o algoritmo proposto apresenta pior precisão enquanto no cenário 3 apresenta melhores resultados nas três métricas consideradas.

Estes resultados penso ser devido às imagens de profundidade, que no caso do cenário 2 apresenta muito ruído que aumenta a quantidade de falsos positivos.

O detetor de face, como era de esperar, segundo o estudado na revisão bibliográfica, apresenta baixa taxa de deteção devido às grandes distâncias, aos movimentos que colocam as pessoas de costas ou de perfil para a câmara e à baixa resolução da câmara RGB.

O detetor com melhor prestação é o detetor de "Ombro-Cabeça-Ombro", sendo também o que apresenta menor variação de valores entre a aplicação apenas dos detetores e o algoritmo proposto. Este detetor apresenta bons resultados na deteção com algum nível de ocultação devido às ocultações presentes serem maioritariamente presentes na parte inferior do corpo. Como acontece com os detetores de HOG e de face, a taxa de deteção é menor com a aplicação do algoritmo proposto, mas apresenta um aumento na fiabilidade das deteções devido à redução dos falsos positivos pela utilização de áreas de interesse em vez da imagem completa.

Em relação à subtração de fundo, verifica-se que o algoritmo apresentado obtém boas "deteções" de pessoas, pois os movimentos nas sequências utilizadas é produzido apenas por pessoas. A subtração não é um detetor de pessoas, mas como os movimentos presentes são produzidos apenas por pessoas, é utilizado o *Ground Truth* e as mesmas métricas que nos detetores para quantificar o erro da subtração. Com uma análise comparativa entre a subtração de fundo implementada e o método MOG verifica-se que a subtração implementada apresenta melhores resultados devido à utilização de informação de profundidade fundida com a imagem RGB. A subtração de fundo implementada consegue detetar diferenças nas imagens e não adiciona ao modelo de fundo uma pessoa ou objeto quando este se apresenta imóvel por um período de tempo. No entanto, o algoritmo desenvolvido apresenta limitações: no inicio do algoritmo é necessário que as primeiras imagens representem o cenário sem pessoas.

Os objetivos referentes à deteção de pessoas com algum nível de ocultação, testes de influência da luz aos dados de profundidade do *Kinect* e segmentação da imagem utilizando subtração de fundo foram cumpridos. A determinação da posição da pessoa no espaço 3D não foi implementada devido a problemas nas capturas efetuadas, com o propósito de implementar um sistema de localização.

6.2 Trabalho Futuro

Para trabalho futuro, de forma a melhorar o algoritmo proposto e ampliar a capacidade do mesmo fica o melhoramento do algoritmo. O algoritmo de subtração de fundo implementado apresenta uma limitação devido ao modelo de fundo da imagem de profundidade não ser atualizado. Uma sugestão para resolução deste problema é utilizar de um sistema de quantificação com o qual um objeto detetado pela subtração de fundo recebe um valor que vai decaindo ao longo do tempo que este objeto se encontre imóvel. Se esse valor chegar a zero, o objeto é considerado parte do fundo (atualizando o modelo de fundo para imagem de profundidade), se este objeto for classificado como pessoa ou apresentar algum movimento é feito um *reset* do valor atribuído para o máximo.

Como uma das ideias que não chegou a ser executada ficou a localização de pessoas no espaço 3D utilizando a informação da profundidade e da informação RGB com calibração manual, para ficar a conhecer de localização utilizando a imagem RGB.

O algoritmo descrito já apresenta sistema de alinhamento das imagens e uma proposta de filtragem do ruído da imagem de profundidade, mas de forma a melhorar o sistema fica a ideia de implementação do detetor de cantos, discutido na secção 3.2.2 para efetuar o alinhamento de forma automática, e a filtragem do ruído da imagem de profundidade utilizando o contorno da imagem RGB como forma de determinar a que profundidade corresponde a zona de ruído.

Com a análise dos resultados da aplicação dos detetores na imagem RGB sem subtração de fundo, é sugerida a ideia de testar uma alteração do algoritmo proposto de forma a utilizar a subtração de fundo descrita para eliminar os falsos positivos. Desta forma, a subtração de fundo não reduzia a área da imagem a processar mas apenas ajudava na filtragem das deteções.

Para o aumento da taxa de deteção fica a ideia do uso de imagens HD, pelo menos para comparação de número de deteções e a implementação de um sistema de seguimento. A aplicação de algoritmo de seguimento permitindo estimar a posição da pessoa dentro da sala mesmo sem a deteção da mesma.

Por fim, fica a ideia da implementação do algoritmo em sistema de *Real-time*. Como forma de implementação deve ser considerado a exportação do código que está a ser executado em CPU para GPU, visto que o GPU consegue obter melhor desempenho, principalmente na computação gráfica.

Anexo A

Anexos

Nos anexos estão presentes diagramas de blocos referentes ao programa criado para o algoritmo proposto e fluxogramas referentes as duas subtrações referenciadas em 4.2.2 e 4.2.3.

A.1 Fluxogramas

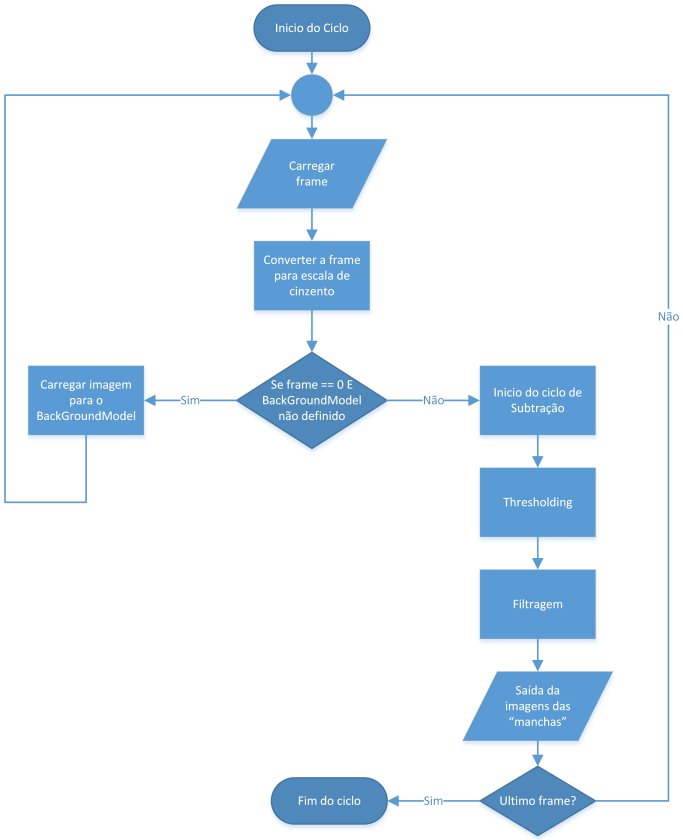


Figura A.1: Subtração de Fundo algoritmo geral.

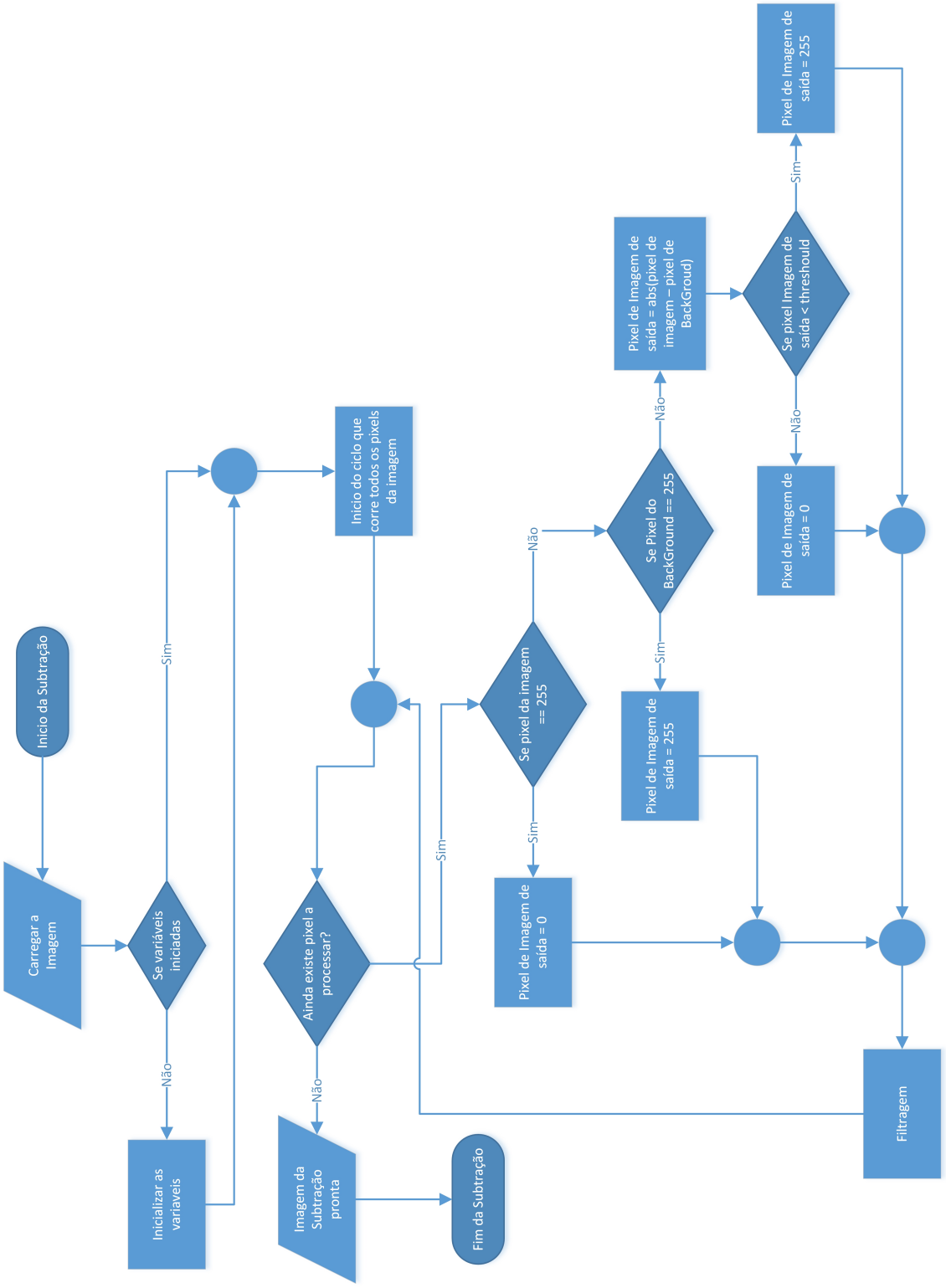


Figura A.2: Algoritmo Subtração de Fundo por diferença entre imagens.

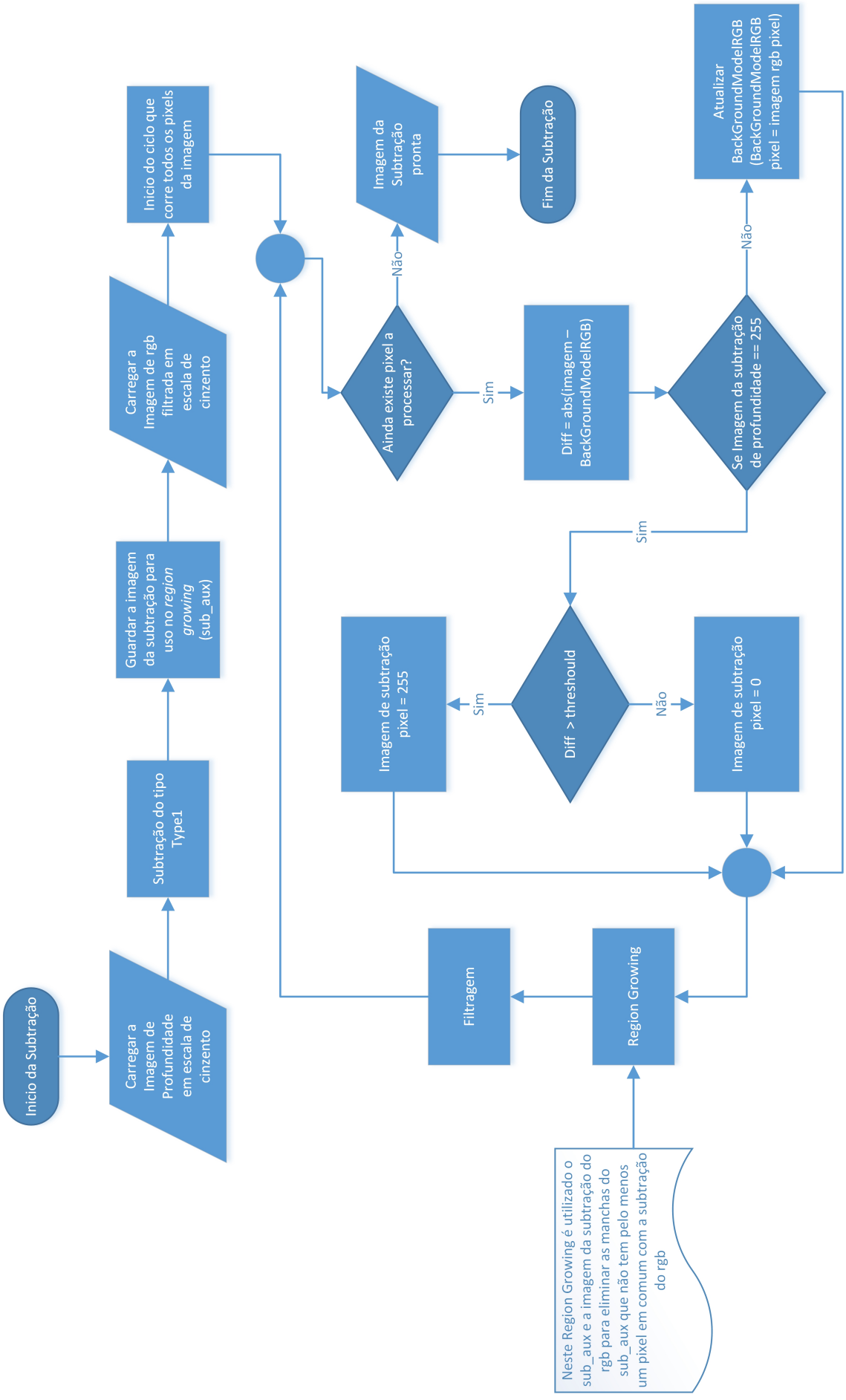


Figura A.3: Algoritmo Subtração de Fundo com combinação de informação.

A.2 Diagrama de Classes

BACKGROUND

```
+BackGroundSub(Mat* backgroud_p, Mat* backgroud_rgb, int Type_chose = 2) : Mat
+BackSubMOG(Mat* img) : Mat
+SetBackGroundType1(Mat* backgroud_p) : void
+SetBackGroundType2(Mat* backgroud_p, Mat* backgroud_rgb) : void
+GetBackGround() : Mat
+GetBackGroundDEPTH() : Mat
+GetBackGroundRGB() : Mat
+GetSub() : Mat
+BackSubType1(Mat* img, int threshold = 5) : Mat
+BackSubType2(Mat* depth, Mat* rgb, int threshold_depth = 10, int threshold_rgb = 10) : Mat
+Filter_background(Mat* img, int median_size = 11, int erode_size = 5, int dilate_size = 15) : void
+SeedSeg(Mat* depth_sub, Mat* rgb_sub) : Mat
```

Figura A.4: Classe BACKGROUND

IMAGEFILTER

```
+GetDepthFilter() : Mat
+GetRGBFilter() : Mat
+ImagesLoad(Mat* depth, Mat* rgb, int mode = 1) : void
+ImagesAlign() : void
+ImagesAlign(Mat* depth, Mat* rgb) : void
+DepthAlign(Mat* img) : void
+ColorAlignDepth(Mat* img) : void
+FilterDepth(int Type = 1) : void
+FilterDepthMat(int Type, Mat* img) : void
+Filter_Depth_Type1(Mat* img) : void
+Filter_Depth_Type2(Mat* img) : void
```

Figura A.5: Classe IMAGEFILTER

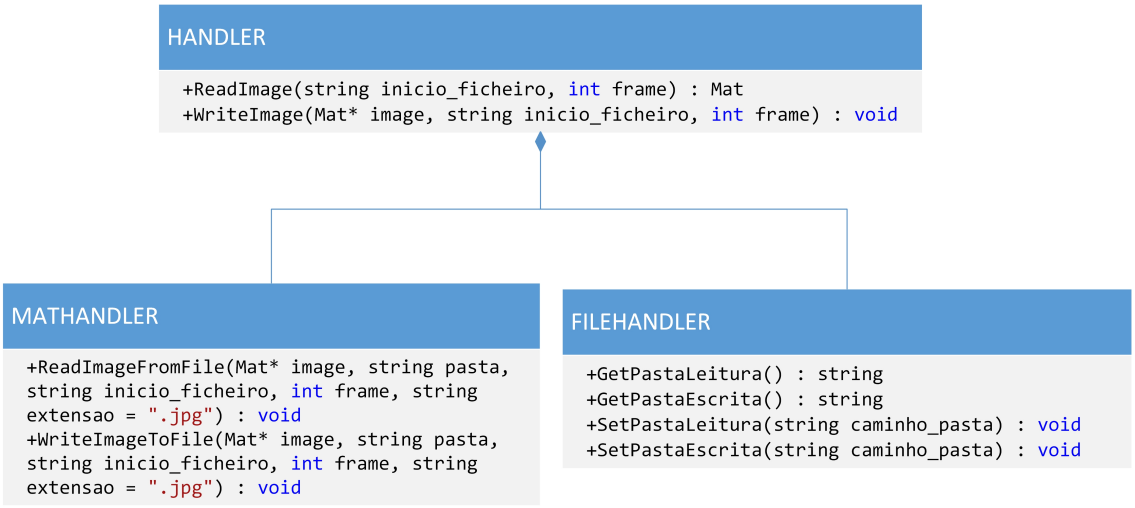


Figura A.6: Classe HANDLER

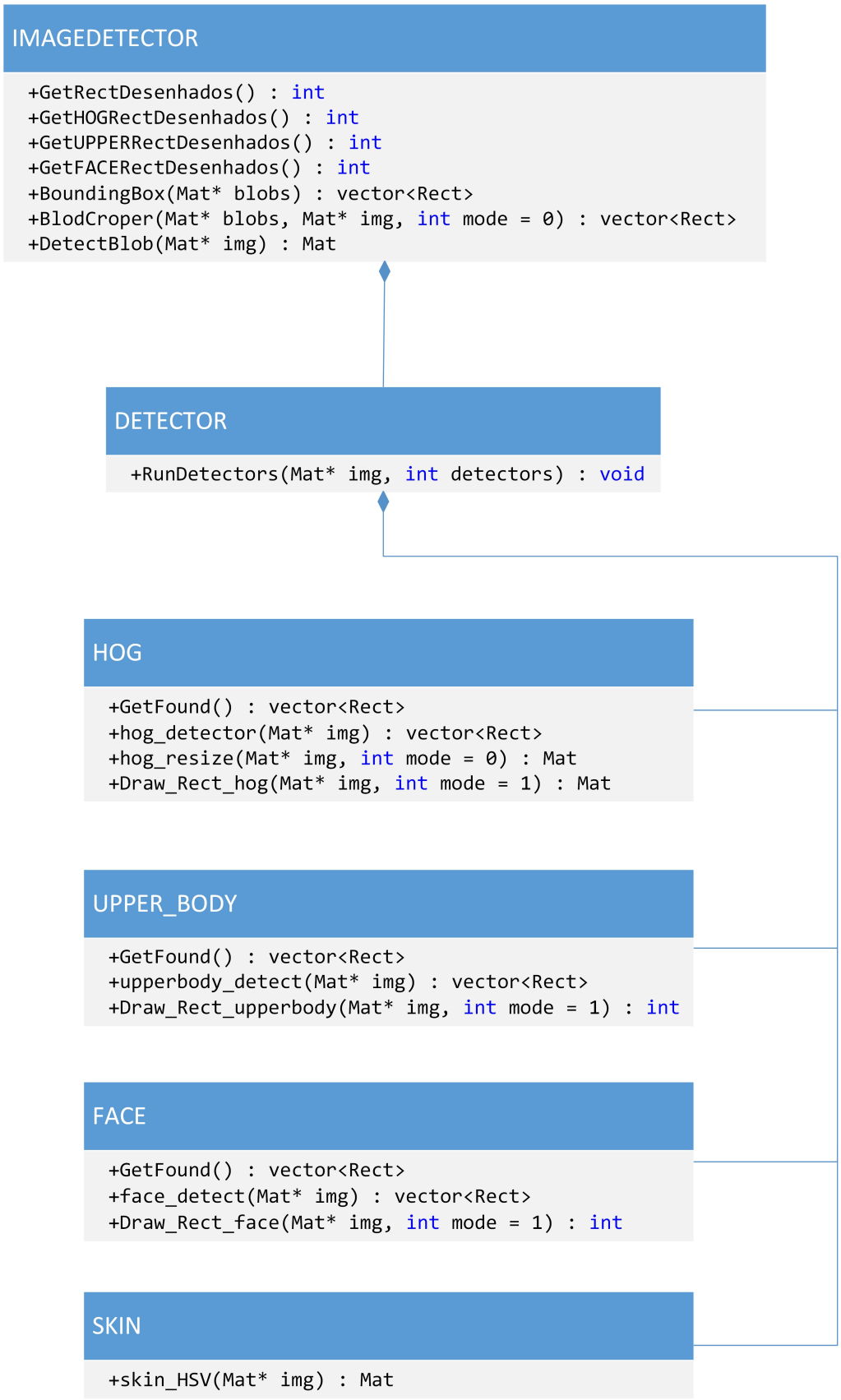


Figura A.7: Classe IMAGEDETECTOR

Referências

[1] HelpAge, 2013. URL: http://www.helpage.org/resources/ageing-data/.

[2] HelpAge, 2013. URL: http://www.helpage.org/global-agewatch/population-ageing-data/country-ageing-data/?country=Portugal.

[3] AMBIENT ASSISTED LIVING JOINT PROGRAMME. Ambient assisted living joint programme catalogue of projects 2013, 2013. URL: http://www.aal-europe.eu/wp-content/uploads/2013/10/AALCatalogue2013_Final.pdf.

[4] AAL, 2012. URL: http://www.aal-europe.eu/about/objectives/.

[5] Mary Belis, 2014. URL: http://inventors.about.com/library/inventors/bldigitalcamera.htm.

[6] PennState, 2014. URL: https://www.e-education.psu.edu/lidar/l1_p4.html.

[7] R. Koch A. Kolb, E. Barth e R. Larsen. Time-of-flight cameras in computer graphics. *COMPUTER GRAPHICS forum*, 29:141–159, 2010.

[8] L. Cruz, D. Lucio, e L. Velho. Kinect and rgbd images: challenges and applications. Em *2012 XXV SIBGRAPI Conference on Graphics, Patterns and Images Tutorials (SIBGRAPI-T), 22-25 Aug. 2012*, 2012 XXV SIBGRAPI Conference on Graphics, Patterns and Images Tutorials (SIBGRAPI-T), páginas 36–49. IEEE Computer Society. URL: http://dx.doi.org/10.1109/SIBGRAPI-T.2012.13, doi:10.1109/SIBGRAPI-T.2012.13.

[9] Inc. Point Grey Research, 2014. URL: http://ww2.ptgrey.com/stereo-vision/bumblebee-2.

[10] Microsoft Kinect, 2014. URL: http://www.microsoft.com/en-us/kinectforwindows/discover/features.aspx.

[11] Microsoft. Kinect for windows sensor components and specifications, 2014. URL: http://msdn.microsoft.com/en-us/library/jj131033.aspx.

[12] Xing Guansheng, Tian Shuangna, Sun Hexu, Liu Weipeng, e Liu Huawang. People-following system design for mobile robots using kinect sensor. Em *2013 25th Chinese Control and Decision Conference (CCDC 2013), 25-27 May 2013*, 2013 25th Chinese Control and Decision Conference (CCDC 2013), páginas 3190–4. IEEE.

[13] Xia Lu, Chen Chia-Chih, e J. K. Aggarwal. Human detection using depth information by kinect. Em *Computer Vision and Pattern Recognition Workshops (CVPRW), 2011 IEEE Computer Society Conference on*, páginas 15–22. doi:10.1109/CVPRW.2011.5981811.

[14] ASUS Xtion Pro, 2013. URL: http://www.asus.com/pt/Multimedia/Xtion_PRO_LIVE/#specifications.

[15] SoftKinectic, 2014. URL: http://www.softkinetic.com/Store/tabid/579/ProductID/2/language/en-US/Default.aspx.

[16] Inc. Point Grey Research, 2014. URL: http://yuriythebest.g0dsoft.com/Bumblebee2%20Price%20List%2008-18-06.pdf.

[17] Ronan O'Malley, Edward Jones, e Martin Glavin. Detection of pedestrians in far-infrared automotive night vision using region-growing and clothing distortion compensation. *Infrared Physics and Technology*, 53(6):439–449, 2010.

[18] C. S. Sanoj, N. Vijayaraj, e D. Rajalakshmi. Vision approach of human detection and tracking using focus tracing analysis. Em *2013 International Conference on Information Communication and Embedded Systems (ICICES 2013), 21-22 Feb. 2013*, 2013 International Conference on Information Communication and Embedded Systems (ICICES 2013), páginas 64–8. IEEE. URL: http://dx.doi.org/10.1109/ICICES.2013.6508394, doi:10.1109/ICICES.2013.6508394.

[19] Kelson RT Aires, Andre M Santana, e Adelardo AD Medeiros. Optical flow using color information: preliminary results. Em *Proceedings of the 2008 ACM symposium on Applied computing*, páginas 1607–1611. ACM.

[20] Alex Leykin e Riad Hammoud. Pedestrian tracking by fusion of thermal-visible surveillance videos. *Machine Vision and Applications*, 21(4):587–595, 2010.

[21] S. Ikemura e H. Fujiyoshi. Human detection by haar-like filtering using depth information. Em *Pattern Recognition (ICPR), 2012 21st International Conference on*, páginas 813–816.

[22] Ana Carolina Lorena e André CPLF de Carvalho. Uma introdução às support vector machines. *Revista de Informática Teórica e Aplicada*, 14(2):43–67, 2007.

[23] Trastem Giken, 2009. URL: http://www.trastem.co.jp/eng/palossie_01.html.

[24] Point Grey, 2012. URL: http://ptgrey.com/products/censys3d/censys3d.pdf.

[25] M. Van den Bergh, D. Carton, R. de Nijs, N. Mitsou, C. Landsiedel, K. Kuehnlenz, D. Wollherr, L. Van Gool, e M. Buss. Real-time 3d hand gesture interaction with a robot for understanding directions from humans. Em *RO-MAN, 2011 IEEE*, páginas 357–362. doi:10.1109/ROMAN.2011.6005195.

[26] Paul Viola e Michael Jones. Robust real-time object detection. *International Journal of Computer Vision*, 4:34–47, 2001.

[27] Gavrila. The chamfer system, 2006. URL: http://www.gavrila.net/Research/Chamfer_System/chamfer_system.html.

[28] L. Spinello e K. O. Arras. People detection in rgb-d data. Em *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*, páginas 3838–3843. doi:10.1109/IROS.2011.6095074.

[29] N. Dalal e B. Triggs. Histograms of oriented gradients for human detection. Em *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, páginas 886–893 vol. 1, 2005. doi:10.1109/CVPR.2005.177.

[30] C. Tonelo, A. P. Moreira, e G. Veiga. Evaluation of sensors and algorithms for person detection for personal robots. Em *e-Health Networking, Applications and Services (Healthcom), 2013 IEEE 15th International Conference on*, páginas 60–65. doi:10.1109/HealthCom.2013.6720639.

[31] Timo Ojala, Matti Pietikäinen, e David Harwood. A comparative study of texture measures with classification based on featured distributions. *Pattern Recognition*, 29(1):51–59, 1996. URL: http://www.sciencedirect.com/science/article/pii/0031320395000674, doi:http://dx.doi.org/10.1016/0031-3203(95)00067-4.

[32] Mu Yadong, Yan Shuicheng, Liu Yi, T. Huang, e Zhou Bingfeng. Discriminative local binary patterns for human detection in personal album. Em *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, páginas 1–8, 2008. doi:10.1109/CVPR.2008.4587800.

[33] Wang Xiaoyu, T. X. Han, e Yan Shuicheng. An hog-lbp human detector with partial occlusion handling. Em *Computer Vision, 2009 IEEE 12th International Conference on*, páginas 32–39. doi:10.1109/ICCV.2009.5459207.

[34] OpenKinect, 2012. URL: http://openkinect.org/wiki/Main_Page.

[35] OpenNI, 2013. URL: http://www.openni.org/about/.

[36] OpenCV, 2014. URL: http://opencv.org/.

[37] Khronos Group, 2014. URL: https://www.khronos.org/opencl/.

[38] R. B. Rusu e S. Cousins. 3d is here: Point cloud library (pcl). Em *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, páginas 1–4. doi:10.1109/ICRA.2011.5980567.

[39] PCL. URL: http://pointclouds.org/about/.

[40] Microsoft. Kinect constants, 2014. URL: http://msdn.microsoft.com/en-us/library/hh855368.

[41] Microsoft. Coordinate spaces, 2014. URL: http://msdn.microsoft.com/en-us/library/hh973078.aspx.

[42] Zhang Zhengyou. Microsoft kinect sensor and its effect. *MultiMedia, IEEE*, 19(2):4–10, 2012. doi:10.1109/MMUL.2012.24.

[43] Eric Gregori. Build smart robots, 2011. URL: http://buildsmartrobots.ning.com/profiles/blogs/one-year-anniversary-for-the-kinect-over-10-million-units-shipped.

[44] Bernardin Keni e Stiefelhagen Rainer. Evaluating multiple object tracking performance: the clear mot metrics. *EURASIP Journal on Image and Video Processing*, 2008, 2008.

[45] Faisal Bashir e Fatih Porikli. Performance evaluation of object detection and tracking systems. Em *IEEE International Workshop on Performance Evaluation of Tracking and Surveillance (PETS)*, volume 5.

[46] Pakorn KaewTraKulPong e Richard Bowden. *An improved adaptive background mixture model for real-time tracking with shadow detection*, páginas 135–144. Springer, 2002.

[47] M. Piccardi. Background subtraction techniques: a review. Em *Systems, Man and Cybernetics, 2004 IEEE International Conference on*, volume 4, páginas 3099–3104 vol.4. doi:10.1109/ICSMC.2004.1400815.

[48] Chris Stauffer e W Eric L Grimson. Adaptive background mixture models for real-time tracking. Em *Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on.*, volume 2. IEEE.

[49] OpenCV. Mog tutorial, 2011-2014. URL: http://docs.opencv.org/trunk/doc/tutorials/video/background_subtraction/background_subtraction.html.

[50] OpenCV. Face detection using haar cascades, 2011-2014. URL: http://docs.opencv.org/trunk/doc/py_tutorials/py_objdetect/py_face_detection/py_face_detection.html.

[51] VA Oliveira e A Conci. Skin detection using hsv color space. Em *H. Pedrini, J. Marques de Carvalho, Workshops of Sibgrapi*, páginas 1–2.